SUC-SESSION

- News
- ZX-Team Treffen in Helmarshausen
- Retro-Programmierung mit modernen Mitteln
- Ein Blick unter die Gummimatte
- »Wilde Pixel«
- Alternative zum Microdrive - ZXPicoMD
- Und vieles mehr ...

DIE

235.

AUSGABE IHRES MAGAZINS

DAS MAGAZIN FÜR DEN SINCLAIR ZX SPECTRUM

VORWORT

Hallo Spectrum-Freunde,

das ich heute das Vorwort schreiben darf, hat leider einen nicht ganz so erfreulichen Grund. Unser Chefredakteur und Magazin-Macher Joachim hatte eine Operation an seiner Hand und kann daher schlicht nicht tippen. Wenn alles Pech zusammenkommt, dann sicherlich auch dieses außergewöhnliche Sommerloch, in dem nun schon Joachim als Schreiber weitgehend ausfiel, aber auch viele andere aus beruflichen und privaten Gründen nicht wie gewohnt, erhoffte oder zugesagte Texte lieferten. Zur Stunde kann ich noch nicht sagen wie die endgültige Ausgabe aussehen wird, den Joachim kann im Moment nicht am Heft arbeiten und wird dies erst mal auch nur schleppend tun. Dennoch bin ich mir sicher, dass er ein tolles Heft zusammenstellen wird.

Trotzdem: Das Heft ist von Euch und für Euch. Am Ende müssen nicht wir Redakteure das Heft erstellen, sondern ihr Leser sorgt selbst dafür. Der Beitrag für die PDF Ausgabe von 4,50 Euro im Jahr deckt laufende Kosten des Clubs, es reicht aber leider nicht für unsere SUC-Privatjacht oder den Jet, insofern können wir Euch nicht die Arbeit abnehmen und das ganze Heft alleine machen. Tatsächlich haben wir das in der Vergangenheit sehr häufig gemacht, in der Ausgabe 233 z.B. waren 8 von 14 Artikeln von uns, einer von uns übersetzt und das Vorwort, Leserbriefe und News in Kürze stammen auch weitestgehend aus unserer Feder. Das ist am Ende zu viel und das werden wir nicht mehr leisten können.

Es hatte sich viel getan, die Adventure Ecke wurde seitdem abgegeben und die Spiele-Reviews sowieso schon länger. Aber Beides findet ihr in dieser Ausgabe nicht, weil die Texte ausgeblieben sind. Das ist nicht weiter schlimm, es ist sogar gut, wenn noch mehr merken, welcher Aufwand hinter unseren Texten steht. Dazu kommt noch das Gestalten der Texte in ein Heft, das Verwalten der Mitgliederdaten, der Druck und das Versenden des Heftes bzw. der PDF-Mails und die Planung drum herum.

Versteht mich nicht falsch, wir machen das. Es wird auch immer wieder Artikel von uns geben, so auch in diesem Heft. Aber im Wesentlichen ist unsere Aufgabe das Heft selbst zu produzieren, nicht die Artikel. Wenn ihr also ein volles Heft haben wollt, dann schiebt es nicht auf die lange Bank, nicht auf nächste Woche, nicht mal auf morgen. Fangt sofort an. Wir brauchen Euch. Wir brauchen Eure Artikel, wir brauchen Euer Feedback. Jeder kann hier ein Programm testen, jeder kann eine neue Hardware, die er sowieso gerade erstanden hat, vorstellen. Jeder kann zumindest uns ein Feedback geben und am Ende kann so ein Puzzlestück liefern, das wir dann in ein ganzes Heft verwandeln. Scheut auch nicht auch eine einzelne Programmvorstellung zu senden. Habt ihr ein Lieblingsspiel? Oder gar irgendeine rare Hardware die noch nicht vorgestellt wurde? Es muss nicht immer was neues sein, gerade wir Spectrum-User entdecken doch »das Alte« immer wieder neu. Es ist faszinierend, was es alles schon gab, aber irgendwie wieder verschwunden ist. Hier gibt es jede Menge über das wir schreiben können.

Ich bin mir sicher, wir werden jetzt überflutet mit Artikeln. Das macht uns Arbeit, aber glaubt uns, wir freuen uns darauf.

Viel Spaß mit diesem Heft und besondere Grüße an Joachim, der jetzt dieses Vorwort ins Heft setzen darf.

Thomas

INHALT

Vorwort, Inhalt, Impressum 2
Wie abonniere ich die *SUC-Session* 3
Leserbriefe 4
Neuer in Kürze 5
25. Spectrumania 7
ZX-Team Treffen in Helmarshausen 8
Retro-Programmierung mit modernen Mitteln - Teil 2 9
Ein Blick unter die Gummimatte 10
Wilde Pixel 15
Das ZXPicoMD 18
Dizzy 4 – Magicland - Spielelösung 19
Lösung Puzzlerätsel 22
Tastaturfolien für Retro-Computer 23

IMPRESSUM

Das **SUC-Session**-Magazin erscheint in Deutsch und in Englisch und kann beim **SpectrumUserClub** abonniert werden.
Infos hierzu bei Thomas Eberle.

Redaktion SUC-Session 234:
Joachim Geupel & Thomas Eberle

Redakteur SCENE+: Thomas Eberle

Clubleitung:
Mirko Seidel, Adresse: Birkenweg 8
89197 Weidenstetten
email: mirko.seidel1@gmail.com

Mitarbeiter an diese Ausgabe sind:

Joachim Geupel	Joa.Geu@gmx.de
Thomas Eberle	thomas.eberle@sintech-shop.de
Harald Lack	
Christof Odenthal	
Heinz Junek	
Norbert Opitz	

Die Rechte der Artikel sowie der Bilder und Grafiken liegen bei den einzelnen Autoren, Fotografen, Zeichnern und Programmierern.

Titelbild von Website *ZXART.EE*

Wir danken allen Beteiligten für ihre Beiträge!

Redaktionsschluss für SUC SESSION 236
Sonntag, 5. November 2023

WIE ABONNIERE ICH DIE SUC-SESSION

Generell ist das Magazin die Hauptleistung des SPECTRUM-USER-CLUBS. Das Heft wird von Spectrum-Usern und für Spectrum-User gemacht. Jeder kann mitmachen. Durch die Mitarbeit kann man sich das Heft verdienen.

Das Heft wird in Abos â drei Ausgaben angeboten. Üblicherweise erscheinen drei Ausgaben im Jahr.

Abo 1:
Drei Ausgaben als gedruckte Ausgabe
Preise:

Deutschland:	22,00 €
EU-weit:	26,00 €
Weltweit:	29,00 €

Abo 2:
Drei Ausgaben als PDF
Preis weltweit: 4,50 €

Geplant ist mit dem Abo auch ein exklusiver Zugang zu Inhalten nur für die Leser, Software und ältere PDF Ausgaben der SUC-Session zum Download. Solange die Webseite dafür noch nicht steht, werden diese Inhalte auf Wunsch per Email gesendet.

Der Club hat folgende Bankverbindung, (nur diese Daten nutzen, nicht die Daten der Fa. Sintech!):
IBAN: DE93630500000000719102,
BIC: SOLADES1ULM,
Kontoinhaber: Mirko Seidel
PAYPAL-Zahlung ist ebenfalls möglich:
https://www.paypal.me/Micro256
oder über unsere Webseite:
www.speccy-scene.de

Selbstverständlich kannst Du auch andere Zahlungswege wie Geld im Einschreiben, Scheck oder sonstiges nutzen. Auch eine E-Mail oder Telefonanruf wegen den Adressangaben wäre wichtig.

Statt in Euro kann man auch in SUC-Punkten bezahlen, 1 Punkt ist 1 Euro wert. Es gibt folgende Punkte:

1) Artikel einsenden **2 Punkte**, aktiviert bei Veröffentlichung
2) Artikel übersetzen deutsch-englisch oder englisch-deutsch **1 Punkt**
3) Anfragen und kleine Leserbriefe bekommen vorerst keine Punkte, wer aber einen Leserbrief sendet, der als richtiger Artikel verwendet werden kann, der bekommt dafür natürlich auch **1 Punkt**.

Bei Interesse an der Mitarbeit auf jeden Fall Kontakt aufnehmen. Wir verteilen Übersetzungsarbeiten und versuchen zu ordnen, damit nicht zwei Autoren an demselben Thema arbeiten.

Wer bereits ein Abo hat, wird von uns angeschrieben wenn das Abo ausläuft. Bitte dann immer gleich reagieren, damit wir rechtzeitig die Mengen für das nächste Heft planen können.

LESERBRIEFE

Danke wie immer für euer Feedback. Für uns ist es immens wichtig auch von euch Hinweise zu bekommen, was gut war oder was wir besser machen können. Bitte schreibt eure Kommentare am besten immer gleich wenn sie euch einfallen an:

thomas.eberle@sintech-shop.de

Simone Voltolini **aus San Giorgio in Italien kann es leider immer noch nicht fassen:**

No disc anymore? ☹

Hallo Simone,

um ehrlich zu sein, war die SCENE+ wirklich ein Teil der Arbeit hier in der SUC-Redaktion, die besonders viel Spaß gemacht hat. Man musste die Programme hacken, um diese auf Diskette zu bringen, diese auch testen und dann auch für die verschiedenen Disksysteme konvertieren. Die meiste Zeit, die ich in meinem Leben mit programmieren verbracht habe, habe ich wohl damit verbracht, Routinen umzuschreiben damit die Diskettenversion läuft. Sehr viele tolle Programme sind auch mit der Hilfe anderer auf Diskette konvertiert worden und dieses Vermächtnis soll nicht untergehen. Vielmehr wollen wir die Inhalte digital präsentieren zum Herunterladen. So weit ist es noch nicht, aber das wird noch.

Glücklicherweise kam ein Lebenszeichen von ***Ellvis*** **aus Berlin:**

Hi Thomas! Danke für das letzte Magazin und deine Nachricht, ich weiß ich hätte mich früher melden sollen. Die zweite Jahreshälfte 2022 war hart für mich und ich war völlig ausgebrannt auf der Arbeit. Im Januar erhielt ich dann eine gesundheitlich schlechte Diagnose und stoppte daher alle meine Aktivitäten. Nun sieht es etwas besser aus, ich bin aber im Begriff meine Arbeitsstelle zu wechseln. Ich habe immer gerne die Spieletestes geschrieben, aber mir fehlt im Moment die Zeit. Ich werde noch einige Zeit brauchen um mich neu zu organisieren, aber das wird im Moment noch dauern.

Viele Grüße, Ellvis

Hallo Ellvis,

schön von dir zu hören. Natürlich ist die Rubrik Spiele-Reviews für uns eine ganz wichtige, wir haben ja das Thema irgendwann zu den tagesaktuellen Programmen aufzuschliessen. Aber es gibt wichtigeres, in erster Linie die Gesundheit. Wir wünschen dir alles Gute auf dem Weg der Besserung und freuen uns wieder von dir zu hören. Wann immer du ein Spiel testen willst, du das gerne. An dieser Stelle aber auch der Aufruf an andere: Ellvis schafft es im Moment nicht. Es muss kein Artikel sein, gerne könnt ihr uns einen einzelnen Spieletest zuschicken. Stand zuletzt war in Ausgabe 234 einige Spiele von 2021 (ich denke noch nicht alle Guten).

Eine Mail erhielten wir von ***Roger Boesch*** **aus der Schweiz. Roger will sich mehr in das Magazin einbringen und Artikel schreiben. Dazu hat er Fragen:**

Hallo Thomas,

ich weiß nicht genau wie ihr das organisiert, aber für mich persönlich wäre es einfacher auch mal einen Artikel zu schreiben, wenn ich das »lange« vorher wüsste und besser noch ein Schwerpunkt Thema vorliegt.

Besser als der Aufruf zu Artikeln wäre für mich deshalb eine Art Media Plan (zugängliches Google Doc. o. ä.) für die kommenden Ausgaben. (planbarer für mich).

Meine Hauptkenntnisse liegen in der Programmierung und ich könnte mir vorstellen, daraus eine kleine Serie zu machen.

Da ich aber an einigen Projekten beteiligt bin, brauche ich da meistens mehr Vorlauf.

Da die »Sinclairs« *aber immer eine Favoritenrolle bei mir einnehmen, würde ich das eigentlich ganz gerne machen um aktiv zu helfen.*

Um aber konkret zu werden, was ich anbieten könnte zum Thema Programmierung:

Entwickeln eines Games in C für ZX Spectrum und/oder Sinclair QL. Am liebsten als Serie, so dass in vier Ausgaben dann ein »einfaches« Spiel entsteht.

Gib mir Bescheid, ob das etwas wäre und bis wann Du den ersten Teil haben müsstest?

Gruß, Roger

Hallo Roger,

ich verstehe was du meinst, aber so professionell sind wir nicht. Die Mitglieder schreiben Artikel so wie sie Zeit erübrigen können. Vorhaben werden umgeworfen, manche melden sich gar nicht mehr, weil es beruflich gerade stressig ist. Andere sagen ihre Mitarbeit einfach ganz ab. Dafür kommen wiederum Andere dazu. Wir können und wollen uns hier nicht einengen, indem wir irgendwelche Planvorgaben machen. Jeder kann einreichen, was er will, das Heft ist von Usern für User. Wir haben ein paar reguläre Sektionen um die wir uns selber kümmern. Das sind »News in Kürze«, »Games (Softwarereviews)« und auch ein bisschen noch die »Adventure-Ecke«, wobei die Miron übernommen hat. Das war's auch schon, alles andere ändert sich immer und es gibt weder einen Plan noch Vorgaben.

Das Gute daran ist, dass wir immer für neues offen sind. Keiner wird eingeschränkt, jeder kann sich beteiligen.

Deine Artikelserie zur Programmierung würde mich persönlich brennend interessieren, aber ich bin sicher auch andere finden das Thema spannend. Ich werde mich sehr freuen, von dir einen Artikel oder eine Artikelserie zu erhalten. Die Deadline für die nächste Ausgabe läuft in 2 Tagen ab, aber ich hab mit Joachim schon ausgemacht, dass wir nochmal einen Monat verlängern müssen. Wenn du also bis Ende Mai was fertig hast, wäre das super. Wenn nicht, dann kommt es eben ein Heft später. Wir wollen hier niemand hetzen, aber es wäre eine tolle Sache.

Dr. Heinz Junek **aus Potsdam ist erst seit Kurzem SUC-Session Leser, will sich aber einbringen, was wir toll finden.**

Hallo Thomas, bisher habe ich in dieser Sache immer mit Joachim korrespondiert. Für das nächste Heft hätte ich zwei Artikel anzubieten:1. Das ZXPicoMD (1 DINA 4 Seite, in Fortsetzung eines Artikels in SUC233). Dieser Artikel ist schon fertig und liegt möglicherweise bei Joachim. Ich kann ihn aber gern noch einmal schicken.2. Ein programmierbares Joystick für den ZX Spectrum (ca. 2-3 Seiten, muss aber noch geschrieben werden). Falls dafür Interesse besteht, dann sagt mir das bitte.
Gruß, Heinz

Hallo Heinz,
gerade beim Thema Hardware können wir glaube ich nicht genug liefen. Der Spectrum ist und bleibt immer ein Computer auf dem viel Hardware entwickelt wird, wohl auch deshalb, weil Sinclair nicht viel lieferte und auch nicht lange existierte. Also jede Art von Interface sehen wir gerne in unserem Heft, ob als Bauanleitung oder Vorstellung.

Hallo liebe Leser,
zum Schluss melde ich mich hier doch noch zu Wort.
Thomas hat es im Vorwort schon angesprochen: eine anstrengende Zeit liegt hinter mir; ne ganze Menge Frust in dem Verein, in dem ich trainiere, krank, (zum Glück) Urlaub, Hand-OP und zu guter Letzt eine Lungenentzündung, die ich zugegebenermaßen zuerst nicht ernst genommen habe.
Das Schreiben fällt immer noch schwer, weil derzeit mein Handgelenk von der OP steif ist, was sich aber allmählich bessert. Ich bekomme auch wieder Luft und Thomas hat mir glücklicherweise das Vorwort abgenommen, weil ich mich mit meinem Vorwort wieder einmal nicht zurück halten konnte - den Thomas sei's bedankt - außerdem bin ich geschlagene vier Tage an der Tastatur gesessen, es überhaupt einzugeben, bevor es dann doch im Papierkorb verschwunden ist.
Und das mit dem Verein löse ich nach den Sommerferien.
Ich habe versprochen, in diesem Heft eine Aufstellung der existierenden Magazine des Spectrum User Club Wuppertal SPCW *und als Nachfolger die des* Spectrum Userclubs SUC *zu veröffentlichen. Dass das nichts geworden ist, dafür möchte ich mich entschuldigen. Ich habe damit begonnen - dann hat der Ärger im Verein angefangen und meine Motivation war breitbandig weg, überhaupt etwas für irgendjemanden zu machen; mit Ausnahme meine Familie versteht sich. Aus demselben Grund gibt es auch diesmal keine Textbeiträge von mir, abgesehen von diesem. Ich habe im Vorlauf zu diesem Heft ein paar Sachen vorbereitet, die ich in die nächsten Hefte einstreuen werde.*
Thomas und ich gehen Konform dahingehend, als dass wir uns etwas aus der redaktionellen Arbeit zurückziehen werden. Das soll jetzt nicht heißen, dass von uns Garnichts mehr kommt, es heißt aber, dass von uns weniger kommt. Von meiner Seite läuft die Reihe der »Retro-Computer« aus. Ich habe so ziemlich alle Computer abgearbeitet, die sich in meiner Sammlung befinden; mit Ausnahme der 16-Bit-Computer. Von den 8-Bit-Computern fehlen noch die Atari-Rechner der XL und XE-Reihe, die MSX-Computer, welche absolut interessant sind, und die Commodore-Computer C64 und C128. Ebenso fehlt der Sinclair QL, der aber eigentlich schon zu den 16-Bit-Computern gehört. Ich werde die Commodore-Maschinen nicht beschreiben, die kennt schließlich jeder, und für die Atari-Computer gibt es Spezialisten, die sich damit besser auskennen. Diese seien hiermit aufgerufen, ihre Lieblingsrechner vorzustellen. Ich werde noch die MSX-Rechner vorstellen - und das war's dann.
Wie es mit dem Heft weitergehen wird, wird die Zukunft zeigen. Es entsteht von einem Heft zum nächsten neu. Letztendlich hängt es auch und zum Großteil von Eurer Mitarbeit ab.
Gruß,
Joachim

NEWS IN KÜRZE

NEUER NEWSLETTER

»On the Spectrum« heißt ein neuer Newsletter, der geplant alle 4 Wochen erscheinen soll und per e-mail erscheint. Im Inhalt geht es um bekannte Klassiker aber auch Spiele die nicht den großen Bekanntheitsgrad erreicht haben. Ferner Interviews mit Persönlichkeiten rund um den Spectrum, Rückblicke und auch Neuheiten. Es kostet nichts sich zu registrieren:
https://onthezxspectrum.eo.page/onthezxspectrum

CRASH COVERTAPE

Die Covertapes sind wieder da. Diesmal kommen sie aber nicht mit einem Magazin. Sind es dann noch Covertapes? Egal. Jedenfalls hat Fusion für das Crash Magazine bereits zwei Covertapes auf dem Markt. Beide sind schon ausverkauft, werden aber vielleicht neu aufgelegt. Dabei ist sicher hilfreich, wenn man sich als Interessent registriert. Man kann das tun indem man auf den blauen Schriftzug unter der »Sold out«-message klickt.

CRASH Cover Tape #1: Mr. Hair´s a very hairy remix, Mr Hair and the kitty katacombs (Demo) TJ´s cavern capers, 1Bit Mechanistic Music demo
https://fusionretrobooks.com/collections

/crash-magazine/products/crash-cover-tape-1?variant=39723458691166

CRASH Cover Tape #2:
Mecki Winter Hibernation, Savage Pricness 2 Demos, Sum it Sup, Astro Phobos
https://fusionretrobooks.com/collections/crash-magazine/products/crash-cover-tape-2?variant=39775412387934

FTP CLIENT FÜR ZX SPECTRUM

Mit »*SARAH*« kann man sich auf einem FTP einloggen und dort Spectrum Dateien (Games) herunterladen. Wie das Ganze funktioniert, kann man auch in einem Video sehen:

https://www.youtube.com/watch?v=ZMfyZwLZNL4

Das Problem ist natürlich, dass der Spectrum keinen Ethernet Port hat und wie immer auf dem Spectrum gibt es schon mehrere Standards die nicht miteinander kompatibel sind. Zum einen das Spectranet, den Spectrum Next und diese Lösung hier ist für die wenigen die ein MB03+ (nicht MB02+) haben. Es ist wirklich sehr schön, aber nur wenige werden es nutzen. Auch werden derzeit wegen der Chipknappheit keine MB03+ produziert. Dies könnte sich aber wieder ändern und dann könnte SARAH vielleicht für noch mehr interessant werden. Download:

https://hood.speccy.cz/dwnld/SARAH1.0.zip

TOWNSEND GAMES AUF DEUTSCH

Clive Townsend war in letzter Zeit recht fleißig, nicht nur mit Spielen zu der *»Saboteur«*-Figur. Einige, wenn nicht alle, Programme werden derzeit übersetzt und auch in deutscher Sprache auf den Markt gebracht. Inwiefern der Markt dafür sehr groß ist, sei dahingestellt, aber wir konnten schon einige Spiele betrachten und uns mit dem Übersetzer Daniel Roth unterhalten. Das Projekt ist schon sehr weit vorgeschritten, wir berichten wie und so man dann die Spiele erwerben kann.

DER NÄCHSTE SABOTEUR

Noch ein Geheimnis aus der Werkstatt von Clive Townsend: *»Saboteur«* wird demnächst auf dem NEXT erscheinen. Ich konnte mich in die Werkstatt schleichen und mit dem Handy einige Fotos machen. Das ganze sieht sehr flüssig aus, die Grafik gelungen. Es ist aber erkennbar das gute alte Saboteur Spiel exakt wie auf dem Spectrum. Wie exakt genau... wir werden es erfahren wenn es fertig ist.

SPECTRUM EMULIERT CHIP8

Thomas Tombinha aus Hamburg ist gerade dabei einen Emulator für den Spectrum fertigzustellen. Also nicht der Spectrum wird emuliert, sondern der Spectrum emuliert einen anderen Computer, in diesem Fall eine der ersten einfachen Spielkonsolen aus den 70er: Chip8 und auch SuperChip8. Wir werden nach Fertigstellung natürlich näher darüber berichten.

LOAD MAGAZIN, AUSGABE 9/2023

Der *Verein zum Erhalt klassischer Computer*, kurz *VzEkC e.V.*, hat sein neues Club-Magazin herausgegeben. Diesmal geht es schwerpunktmäßig um *»Operating Systems«*. Die Titelthemen umfassen einige Artikel wie »Die Geschichte von Betriebssystemen« und deren Anwendung auf verschiedenen Computersystemen. Weiter sind die Computer von Atari, Apple und der (oder evtl. »die«) unvermeidliche Amiga vertreten. Es werden die Systeme und die Firmengeschichte von Tandy Radio Shack TRS-80, WANG Laboratories, die Geschichte des TA M32-Computers und noch einiges mehr vorgestellt. Einige der Artikel sind von echten Insidern geschrieben worden und recht speziell, da sie sich mit den Eigenschaften ganz spezieller Anwendungen befassen. Natürlich fehlen die Vereinsnachrichten nicht, ebenso wie Berichte über Treffen, so dass *Regionaltreffen Wittlich 2023*, *Retrocomputer-Treff Niedersachsen* und die *Classic Computing 2022*.

Das Heft ist klasse und sehr informativ. Einige Artikel setzen Fachwissen in bestimmten Themenbereichen voraus, was den Lesespaß nicht mindert. Das Heft lohnt sich. (jg)

25. SPECTRUMANIA IN WITTENBERG

Ein Jubiläum steht bevor. Norbert richtet die 25. Spectrumania aus! Herzlichen Glückwunsch!

Pension Alabama und VHS Wittenberg

Einfahrt zur VHS Wittenberg

Im Restaurant »Mediterraneo« in der Schloßstraße 33 habe ich für gemeinsames Abendessen Plätze reserviert, am Freitag ab 18.00 Uhr und am Samstag ab 19.00 Uhr.

Hallo alle miteinander !

ich möchte euch zur 25. Spectrumania in der 06886 Lutherstadt Wittenberg einladen.

Der Termin ist Samstag 02. Sept. und Sonntag 03. Sept. 2023 in der Volkshochschule (VHS) und der Pension »Alabama« und ist fest gebucht.

Die Öffnungszeiten sind Samstag 08.30 bis 19.00 und Sonntag 09.00 bis 16.00 Uhr.

Unser Tagungsort ist wieder die VHS Wittenberg in der Falkstr. 83 / Ecke Zimmermannstraße.

Wir bekommen zwei nebeneinanderliegende Räume zur Nutzung.

Für das leibliche Wohl werde ich ein kaltes Bufett mit belegten Brötchen, Kaffee und Getränke vorbereiten.

Tische und Stühle werden von der VHS gestellt. Mitzubringen sind neben einer großen Anzahl von Sinclair- und anderen 8-Bit-Computern auch Verlängerungskabel, Stromverteiler und viel Lust zum gemeinsamen Computern.

Wenn jemand einen interessanten Vortrag halten möchte (ca. 15 min), darf das gemacht werden. Es sollte nur vorher gesagt werden, falls mehrere so etwas machen möchten, zum Koordinieren.

Die Anmeldung, vor allem Übernachtungsreservierungen, sollte bis etwa zum Juli gemacht werden.

Ich bin im September 2018 in eine andere Wohnung umgezogen auf der anderen Elbseite (B2 Richtung Leipzig), dann nach der Elbbrücke an der zweiten Ampel rechts abbiegen, bis zum Kreisel rein und gleich wieder raus, unter der Unterführung durch, und wenn die Straße geradeaus nicht mehr weiter geht, rechts abbiegen, ich wohne in dem Haus mit dem Motorradladen. Am Hoftor rechts an der Hausecke neben den drei Briefkästen ist meine Klingel. Ich muß zwei Etagen runterkommen, da kein Türöffner vorhanden ist, aber Gegensprechanlage (Sturm klingeln).

Die neue Adresse ist, wo auch die Zimmerschlüssel abgeholt werden müssen:

Norbert Opitz
Marktplatz 4
OT Pratau
06888 Lutherstadt Wittenberg
Festnetz: 03491 - 5052169
Handy: 0175 - 2249575 (nur während der Veranstaltung)

Tschüss, Norbert

ZX-TEAM TREFFEN IN HELMARSHAUSEN

VON THOMAS EBERLE

Bereits zum 26. Mal traf sich das »ZX-Team« in diesem Jahr. Die Location hatte sich geändert, fand es doch die letzten Male im hessischen Mahlerts statt, ging es diesmal in eine Jugendherberge nach Helmarshausen. Ich hatte schon mal vor ca. 20 Jahren ein ZX-Team Treffen besucht, damals sind wir zu dritt angereist und waren die einzigen Spectrum-User, während das ZX-Team aus ZX81 Usern bestand. Mittlerweile kennt man auch einige der ZX81-User näher, denn man trifft sich öfter auf z.B. auf dem Spectrum-Treffen in Wittenberg. Aber auch mehr Spectrum-User besuchen das Treffen öfter und so hat sich das ganze eher schon zu einem Sinclair-Treffen gewandelt, mit ZX81, Spectrum und auch QL.

Auch diesmal reisten wir zu dritt, mein Sohn Noah, mein langjähriger Spectrum-Freund Fred und ich. Die allermeisten Besucher nutzen das all-inklusive Übernachtungsangebot der Jugendherberge, während wir aber nur Tagesbesucher waren. So fuhren wir zwar früh los, kamen aber doch erst kurz vor dem Mittagessen im tiefsten hessischen Hinterland an. So malerisch die Landschaft ist, Autobahnen sind einfach schneller. Beeindruckend war die Location aber doch, ein burgähnliches Gemäuer, riesengroß und mehrere Stockwerke. Nach einigen Gängen fanden wir den Raum und dort waren wir dann doch überrascht, dass trotz des großen Gebäudes der Raum doch etwas klein wirkte. Dies kann aber auch an der Besucherzahl liegen, knapp 60 Besucher drängten sich in diesem Raum, an Verkaufsausstellung war gar nicht zu denken. Mit Mühe und Not ergatterten wir einen halben Tisch, natürlich nur, nachdem Dieter freundlicherweise etwas zusammenrückte. Mein mitgebrachter Flohmarkt hingegen musste ich in dem langen Ganz zuvor aufbauen, zwar kein schlechter Platz, aber so konnte ich nicht zeitgleich Verkäufe managen und an meinem Spectrum sitzen. Es endete damit, dass Fred und ich mit einem Laptop auf dem Schoß im Flur einige Programme durchgingen und Noah am Platz die neuesten Spectrum-NEXT releases ausprobierte. Dies soll keine Kritik am Veranstalter sein, ich möchte nur die engen Platzverhältnisse darstellen, aber manchmal sagen ja Bilder mehr als Worte: Man beachte das Bild der Veranstaltung.

Leider gab es nicht allzu viel Vorträge, einen zum Thema »Sinclair Computer in Filmen« gab es aber doch, war recht amüsant und fand zum Glück in einem anderen Raum statt (den ich für größer und damit auch für geeigneter für das komplette Treffen gehalten hätte).

Die Organisation war aller bestens, es gab im Raum Getränke und drei Mahlzeiten am Tag inklusive Getränken. Auch wenn sich das zunächst günstig anhört, die Kosten wurden umgelegt und betrugen pro Besucher immerhin 30,00 € (+ eventuelle Übernachtungskosten von 50-60 € pro Person). Wer also die günstigen Preise in Wittenberg gewohnt ist, mag zusammenzucken, aber natürlich ist so ein Treffen etwas Einmaliges und es ist sehr gut besucht. Zudem liegt der Raum um Kassel so ziemlich in der Mitte von Deutschland, daher für jeden am Ende doch gut zu erreichen, wenn auch aus dem tiefsten Bayern die Anfahrt trotzdem acht Stunden oder mehr bedeuten kann. Letztlich gibt es kaum einen besseren Ort um ein mehrtägiges Treffen durchzuführen und ich vermute ein größerer Saal hätte leider noch höhere Kosten bedeutet. Es war im Hinblick auf die gerade überstandene Corona-Pandemie aber schon ungewohnt, so viele Menschen auf so engem Raum zusammen zu sehen und auch der Grund warum einige absagten.

Aufgrund des engen Raumes kann ich nicht sagen das ich viel gesehen hätte, aber es war allein eine Freude einen so vollen Raum mit Personen zu sehen, die dasselbe Hobby mit mir teilen und was ich nächstes mal ändern muss ist einfach einen noch größeren Geldbeutel mitnehmen und gleich das volle Programm buchen, den die Zeit ging einfach zu schnell um. Ich freue mich schon auf das nächste Mal.

RETRO-PROGRAMMIERUNG MIT MODERNEN MITTELN - TEIL 2

VON THOMAS EBERLE

Hallo Spectrum-Freunde,

nachdem in der Ausgabe 233 ein Artikel erschien, wie man sich Visual Studio Code auf dem PC einrichten kann, um damit Z80 Assembler zu programmieren, habe ich das auch so eingerichtet. Meiner persönlichen Ansicht nach, die aber von einigen in meinem Umfeld geteilt wird, ist Visual Studio Code eine gute Lösung für die, die sowieso als Programmierer tätig sind und denen der Aufbau und das Arbeiten mit Visual Studio Code bekannt ist. Mir selbst versagten schon die Nerven beim Einrichten und ich habe lange gebraucht, ehe das erste kurze Programm als .Z80 File ausgegeben wurde. Einmal eingerichtet und verstanden ist es sicher eine gute Lösung, ich zeige trotzdem hier noch 2 Alternativen auf, die meine Recherche ergeben hat:

8BIT WORKSHOP IDE

Auf https://8bitworkshop.com/ findet Ihr eine Online – Lösung um verschiedene Systeme in Maschinenencode zu programmieren. Zur Auswahl stehen sehr viele Geräte, von Atari Spielkonsolen über C-64 und CPC bis eben hin zum ZX Spectrum. Sehr gut finde ich die gleichzeitige Emulation, man kann wahlweise also programmieren und auf der rechten Seite des Bildschirms direkt sehen was programmiert wurde.

Leider zeigt sich hier auch eine Schwäche: Es ist mir nicht gelungen eine eigene Routine frei zu entwerfen. Stattdessen konnte ich nur die hier angezeigte Beispielroutine „Hello World“ verändern. Sobald man die ORG Befehle am Anfang oder am Ende verändert hat, hatte ich keine Anzeige mehr auf der rechten Seite und konnte so auch nicht die Richtigkeit meiner Routine testen. Weiterer Minus-Punkt ist die mangelhafte Speicherfunktion. Zwar werden die Routinen im Cache des Rechners gespeichert, sollte man aber mal das CACHE löschen, wäre alles weg. Zwar gibt es auch die Option das ganze als ZIP Datei zu speichern, diese kann man aber nicht zurück laden. Es bleibt die Option über einen Texteditor den Code zu kopieren und zu speichern, aber gut gelöst ist das nicht. Möglicherweise erfolgt aber noch ein Update.

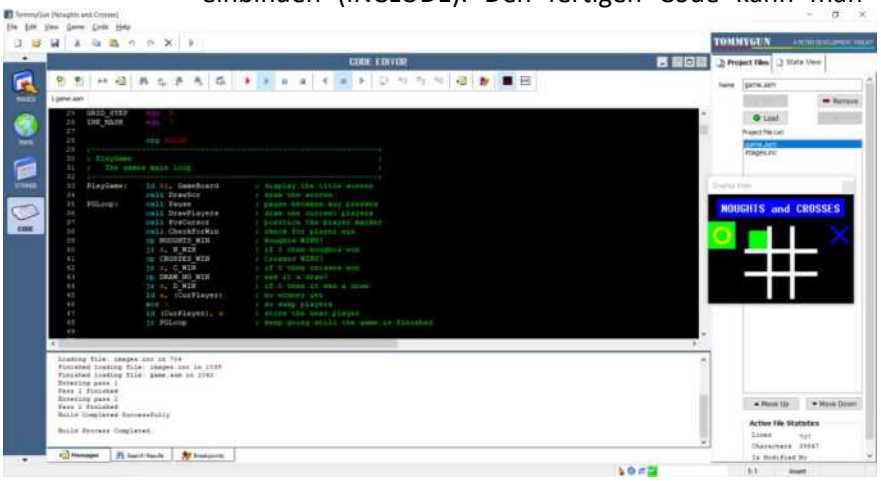
Trotz dieser Schwächen, die Webseite eignet sich wenn man mal unterwegs etwas programmieren will. Man braucht nur Internet-Zugang, sogar auf dem Mobiltelefon könnte man so Routinen entwickeln und dann später kopieren und in anderer Software verwenden. So ist das ganze eher ein Tool, aber keine ganze Entwicklungsplattform.

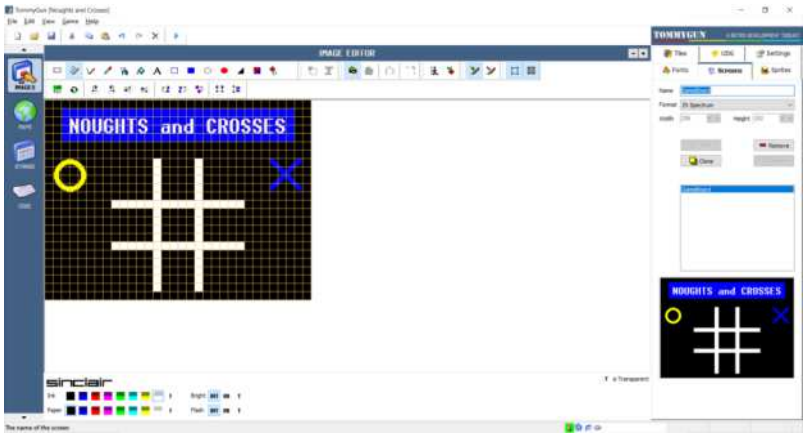
TOMMYGUN

Ganz anders, aber auch nicht ausgereift kommt Tommygun daher. Es funktioniert auf Windows-Rechnern und muss auch installiert werden. Tommygun ist eine ganze Entwicklungsplattform, allerdings hat meiner Meinung nach sich der Entwickler (der Tony und nicht Tommy heißt) sich zu viel aufgehalst. So gibt es sogar eine spezielle Version die besonders mit dem AGD (Arcade Game Designer) von Jonathan Cauldwell zusammenarbeitet. Des weiteren beinhaltet das Programm einen Bereich für Grafik und einen für das Coden. Das ganze ist nicht zwingend nur für den Spectrum angelegt. Tommygun liefert nur eine Oberfläche, man legt selbst an mit welchem Assembler man arbeiten will und auch mit welchem Emulator. So ist Tommygun auch im CPC Bereich bekannt, hauptsächlich verbreitet ist es aber doch bei Spectrum Usern und hier auch optimiert.

Der Grafik Bereich kann Bilder umfassen oder man kann Sprites kreieren. Das arbeiten mit dem Bildschirmeditor empfinde ich nicht so komfortabel als mit Grafikprogrammen, der Sprite-Editor ist aber ganz gut gelöst. Dann hat man die Möglichkeit die Grafik in verschiedenen Formaten zu speichern und damit in den Code einzubinden. Die Programmiersektion ist nichts weiter als ein Texteditor, die Zeilen werden erst beim compilieren mit dem Compiler geprüft. Hier wird die Verwendung von PASMO empfohlen, aber auch SJSAsm ist möglich (für Assembler) oder wer in C programmieren will wählt den Z88DK oder in Basic mit dem Basic Compiler. Wie gesagt, Tommygun lässt hier viel Freiheiten.

Es ist möglich kleine Routinen anzulegen, diese zu speichern und wieder zu laden. Man erstellt ein Hauptfile und kann dann die Nebenroutinen in diese mit einbinden (INCLUDE). Den fertigen Code kann man





dann in einer Art Test-Modus aufrufen, dann wird das erstellte Programm in einem kleinen Nebenfenster angezeigt. Oder man kompiliert und erstellt ein TAP – File, welches dann direkt in einem Emulator gestartet werden kann. Das ist dann für den Endtest.

Eigenartigerweise ist die Möglichkeit Grafik einzuladen bei dem Grafik-Editor anders gelöst als bei dem Code Editor. Man kann aber auch per Copy & Paste Grafiken aus einem anderen Programm einfügen. So kann man das Thema Grafiken mit anderen Programmen zur Bildbearbeitung kombinieren (z.B. SEVENUP) , da Tommygun bessere Möglichkeiten gibt die Bilder dann zu Code zu verarbeiten (im gleichen Format wie der Spectrum-Bildschirmaufbau oder auch Zeile für Zeile). Eleganterweise kann man die Grafiken so erstellen, das bei einer Änderung im Editor auch der daraus erstellte Code sich mit verändert. Wenn man das nicht will, muss man nur die Zeilen »Ressources« im Code der Bilder ändern.

Weiterer Minuspunkt von Tommygun sind die vielen Eigenartigen Meldungen und Abstürze die immer wieder vorkommen. Ganz einfach zum Absturz bringt man das Programm, wenn man ein Projekt öffnet, es schließt und wieder ein anderes öffnet. Man muss also jedes Mal das ganze Programm schließen, wenn man sich ein Projekt anschauen möchte und dann zu einem anderen wechselt.

Die Entwicklung von Tommygun wurde 2014 eingestellt, auch wenn 2017 noch ein kleines Update erschien. Ich denke am Ende hat sich Tony Thompson zuviel aufgehalst, auch ein Musik Programm war noch geplant. Das hätte am Ende also ein Grafik-Programm, ein Musikprogramm und eben eine Programmierumgebung bedeutet, sicher toll wenn es funktioniert, aber was wir im Moment haben ist kein Musikprogramm, ein Malprogramm dem wichtige Funktionen fehlen und ein Editor, der soweit funktioniert und auch das Beste daran ist. Vielleicht hätte man sich darauf konzentrieren sollen.Möglicherweise geht die Entwicklung aber doch irgendwann weiter. Tommygun bekommt ihr z.B. auf:

https://sourceforge.net/projects/tommygun

Insgesamt muss ich sagen, habe ich die Ideal-Lösung noch nicht gefunden. Aber vielleicht habt ihr ja eine. Bitte schreibt dem Club, wie ihr Programme entwickelt. Auch wenn es noch klassisch auf dem Spectrum stattfindet, könnt ihr ja über die Vorteile dieser Programmentwicklung schreiben. Wir hören gerne von euch:

thomas.eberle@sintech-shop.de

(te)

EIN BLICK UNTER DIE GUMMIMATTE

»SPECTRUM INSIDE«

RETROSPEKTIVE: © 2023 HARALD LACK

Verwendete Hardware: Original ZX Spectrum 48K
Veröffentlicht im Spectrum User Club

Liebe 8-Bit (Spectrum)-Freunde!

Der Sinclair ZX 80 wie auch der ZX 81 sowie dessen direkter Nachfolger, der ZX Spectrum sind in einschlägigen 8 Bit-Computerkreisen hinlänglich bekannte Systeme. Nun gibt es unseren *Speccy* ja auch schon über 40 Jahre und die Entwicklungen, die rund um diese Maschine durch die Fans entstehen, sind aus meiner Sicht mehr als einzigartig. Es gehört schon ein profundes Wissen dazu, diese ganzen Neuerungen an ein durchaus in die Jahre gekommenes Computersystem anzupassen. Das vor rund 40 Jahren Neue an diesen Sinclair Computern war Sir Clive´s Hang zur nahezu schon genial zu nennenden Reduzierung von Bauteilen. Hier wurden Kniffe und Tricks angewendet, die selbst uns als langjährigen Spectrum Vertrauten gelegentlich nur mit Mühe durchschaubar sind oder auch für immer unbegreiflich bleiben. Nicht selten gingen die angewandten Tricks weit über das normale Ausnutzen von vorhandenen Komponenten hinaus (speziell der Zilog Z80 sei hier erwähnt, der durchaus »interessanten« Anpassungen unterworfen war). Nahezu jede (versteckte oder bekannte) Eigenschaft irgendeiner Komponente, jedes irgendwo im System vorhandene Signal wurde in ungewöhnlicher und unvorhergesehener Weise verwendet, um dem System benötigte Funktionalitäten zu verleihen. Stellvertretend möchte ich hier nur die Art der Bilderzeugung aufführen, die aus einer fast schon grotesk zu nennenden Zweckentfremdung des Interrupts entstanden ist. Und dann war da auch noch diese mysteriöse *ULA*, ein Baustein, den man so in keinem anderen Computer dieser Zeit vorgefunden hat. Soweit nichts Neues in Spectrum-Kreisen. Und trotzdem bleibt es vielen von uns irgendwie unerklärlich insbesondere dann, wenn man so wie ich kein Technikfreak ist.

In meinem heutigen Rückblick auf den Spectrum, den ich in Anlehnung an das berühmt berüchtigte Intel Logo einfach mal *»Spectrum Inside«* genannt habe (Intel möge mir die Zweckentfremdung verzeihen), will ich eine allgemeine Betrachtung zum Spectrum (hier beschränke ich mich auf das Standardsystem des Spectrum 48 K) geben und ein bisschen über das »*Innenleben*« sprechen. Speziell möchte ich mich ein wenig der wichtigsten Systemkomponente widmen, dem Betriebssystem im Spectrum ROM. Wo angebracht werde ich auch Vergleiche mit dem ZX 81 anstellen um zu versuchen manches klarer verständlich zu machen soweit mir das in meinem laienhaften Kenntnissen möglich ist. Sicherlich ist das nachfolgend Gesagte vielen von uns in irgendeiner Weise geläufig und es wurde so oder in ähnlicher Weise bereits darüber berichtet, dennoch darf ich hier einen kurzen zusammenfassenden Rückblick auf mehr als 40 Jahre *Spectrum-Geschichte* geben, auf die man auch heute noch gelegentlich zurückschaut.

Aber der Reihe nach. Ebenso wie der ZX81 verfügte der Spectrum über den magischen Bauteil einer ULA die kundenspezifischen Anforderungen genügen musste. Damit waren in dieser ULA alle benötigten Funktionen konzentriert, die über den Prozessor sowie RAM und ROM hinausgingen. Der im System verwendete Zilog Z80 Prozessor war mit seinem 8 Bit breiten Bus in der Lage bis zu 64 KByte Speicher zu adressieren. Diese wurden über die vorhandenen 16 Adressleitungen angesprochen. Im Folgenden wollen wir uns ganz auf den Spectrum konzentrieren, da ich selbst keine Erfahrungen mit dem ZX 81 habe und deshalb das Feld hierbei lieber den Spezialisten überlassen möchte um profund darüber zu schreiben.

Wie schon erwähnt, basierte der Spectrum weitestgehend auf seinem unmittelbaren Vorgänger, dem ZX 81. Insgesamt wirkt die Reduzierung der Bauteile beim Spectrum nicht mehr so ausgeprägt wie noch beim ZX 81. Nichts desto trotz lässt die Programmierung noch seine Verwandtschaft erkennen. Hardwaremäßig ist er natürlich etwas aufwendiger konzipiert, liefert er ja jetzt auch ein Farbbild, was dem ZX 81 noch verwehrt war. Die wohl am bekanntesten und sehr verbreitete Version des Spectrum war der so genannte 48K, der - wie der Name schon sagt - mit den für damalige Zeiten immens großen Hauptspeicher von 48 KByte verkauft wurde. Rechnet man noch die vorhandenen 16 KByte für das ROM hinzu, dann war der gesamte für den Z80 ansprechbare Adressbereich von 64 Kbyte vollständig ausgenutzt. Wenn man jetzt noch von den rund 48 KByte RAM die ca. 7 KByte Bildschirmspeicher abzieht, dann kommt man zu den bekannten rund 42 KByte wirklich frei verfügbaren Speicher, den man auch über den integrierten BASIC Interpreter nutzen konnte. Abgefragen kann man das alles über den bekannten Befehl **`PRINT 65535 - USR 7962`**.

Bekannterweise verfügt der Prozessor Z80 über spezielle Befehle (ich habe ja schon mal vor vielen Jahren hier in der Clubzeitung über die Programmierung des Z80 berichtet) und eine eigene Steuerleitung für Input/Output Operationen. Das hat den Vorteil, dass zusätzlich angeschlossene Geräte wie die Tastatur usw. den Speicherbereich nicht belasten, denn durch die IN- und OUT-Befehle sind die gesamten 64 KByte als I/O Adressen oder Ports nutzbar.

Durch den für damalige Zeiten (wir reden von Anfang der 80er-Jahre) großen Speicherbereich (andere Homecomputer wie das gerne auch genannt wurde hatten auch nicht mehr zu bieten) war es beim Spectrum erstmals möglich, einen festen Bildschirmspeicher im System zu definieren. Einen zeichenorientierten Bildschirmspeicher wird man im Spectrum vergeblich suchen obwohl das vielleicht der eine oder anderen von uns vermuten könnte, sind doch Zeichen auf dem Bildschirm sichtbar. Das Zeichensatz im ROM ist dafür verantwortlich, daß der Bildschirminhalt eine grafische Form bekommt. Das war zu seiner Zeit auch ein wesentlicher Unterschied zu anderen Computern. Vielmehr ist der Bildschirm ja bekanntermaßen in Bildpunkten aufgebaut, jeder Punkt ist ein Bit im Speicher.

Was bedeutet dies nun konkret? Ich möchte versuchen es an einem wirklich sehr einfachen Beispiel zu verdeutlichen. Beim Schreiben eines Zeichens auf dem Bildschirm (resultierend aus irgendeinem Tastendruck unsererseits oder einer Ausgabe eines Programmes) wird die punktmäßige Form des darzustellenden Zeichens dem Zeichensatz entnommen und sozusagen als Bild im Speicher abgelegt. Das bedeutet umgekehrt aber auch, dass die Abbildung von Zeichen und grafischer Einzelpunkte im Speicher identisch behandelt werden kann. Deshalb gibt es auch keine unterschiedlichen Bildschirmmodi für Grafik und Zeichensätze. Wenn wir jetzt ein bisschen rechnen, dann kommen wir zu folgenden Resultaten. Der Standardbildschirm des Spectrum besteht ja aus 22 Zeilen zu je 32 Zeichen. Da bekannterweise jedes Zeichen aus einer 8*8 Matrix abgebildet wird, haben wir im Pixelformat dann eine Bildschirmauflösung von 22*8 Punkte je Zeile zu 32*8 Punkte je Spalte oder eben die bekannten 176*256 Pixel des Bildschirms. Wie wir wissen, erlaubt der Spectrum aber auch noch einen Befehl OVER, der eine Überlagerung von Zeichen, ohne dass vorher an dieser Stelle stehende Zeichen zu löschen, ermöglicht. Zusätzlich benutzt der Spectrum Zeichensatz auch noch so genannte Grafiksymbole und unterscheidet sich damit ebenfalls zu anderen Computern nicht nur seiner Zeit. Und als Höhepunkt der Verwirrung gibt es noch die guten alten Attributwerte die in vier Ausprägungen erscheinen (permanent, transparent, temporär und

temporär transparent). Wer hier noch den Überblick bewahrt ist wirklich zu beneiden. Ich jedenfalls war da lange total genervt, aber letztendlich habe ich es dann irgendwann mit einiger Geduld auch verstanden. Doch genug davon.

Kommen wir nun aber zu dem, was einen Computer wirklich ausmacht – dem Betriebssystem. Neben den hardwareabhängigen Gegebenheiten ermöglicht uns erst das Betriebssystem mit dem Computer wirklich in Interaktion zu treten. Nochmals hier zu Klarstellung: Das Betriebssystem ist nicht die Sprache, in der wir mit dem Computer kommunizieren (auch wenn das beim Spectrum vielleicht irrtümlich leicht so verstanden werden kann), sondern der Teil der internen Software, welche die vorhandenen Einheiten des Systems anbindet und verwaltet. Unter diese Einheiten fallen vor allem die Tastatur, Speichermedien, Bildschirmsteuerung, Drucker usw. In der Regel ist dabei das eigentliche Betriebssystem weitestgehend von anderen Speicherbereichen getrennt. Auch bei BASIC Rechnern wie dem Spectrum. Keine Ausnahme macht unser Speccy hier auch in der Hinsicht, dass das Betriebssystem und der Sprach-(BASIC)Interpreter nicht getrennt nebeneinander im ROM stehen. Bekanntlich benutzt der Spectrum eine an das Microsoft BASIC angelehnte Variante dieser Sprache, wenngleich sich insbesondere in der Stringverarbeitung doch einige Unterschiede, auf ich hier aber nicht näher eingehen will, auftun.

Jetzt wollen wir uns aber noch ein paar Einheiten und deren Anbindung an das Spectrum ROM näher ansehen. Beginnen wollen wir mit der Tastatur.

Wer ein disassembliertes ROM Listing des Spectrum Standard ROM vorliegen hat, wird sehen, dass die Tastatur ab der Adresse **$028e HEX** eingebunden wird. Angelehnt an den ZX 81 wurde hier ein wirklicher Meisterkniff an Aufwandsminimierung betrieben. Gemeint ist der **Befehl 029 ED 78 NXTREI IN A,(C).** Durch diesen Befehl werden die B und C Register des Z80 als Adresse ausgegeben. Ebendies ist gleichbedeutend mit der Zeile und Spalte der Matrix der Taste, die gerade angesprochen wird. Dadurch geschieht nichts anderes, als dass der getätigte Tastendruck entsprechend dekodiert wird. Wer sich schon näher mit den einzelnen ROMs der Sinclair Rechner beschäftigt hat, wird feststellen, dass die Routine fast identisch mit der des ZX 81 ist, die interne Codierung sich aber doch in wesentlichen Punkten unterscheidet. Der Spectrum benutzt hier standardgerecht eine genormte ASCII Zeichentabelle, die auch die Groß- und Kleinbuchstaben berücksichtigt. Dies ist umso bemerkenswerter, da beim ZX 81 von Normierung keine Spur zu finden war. Wirklich interessant wird es aber bei unserem Computer erst dann, wenn eine Taste gedrückt wird.

Warum? Nun, die Reaktion auf einen Tastendruck ist ein wirklich interessantes Gebiet, das sich unserer Logik nicht immer gleich erschließt. Ein wirklich sehr kompliziertes System von logischen Abfragen ergibt pro Tastendruck keinesfalls immer einen Buchstaben, denn die Tasten beim Spectrum sind ja bekanntlich mehrfach belegt. Außerdem werden ja auch alle BASIC Befehle / Funktionen im Ein-Tasten-Modus mittels sogenannter Token eingegeben. Wenn man bedenkt, dass abhängig von der durchgeführten Syntaxprüfung und den beiden verschieden wirkenden Shift-Tasten sich bei einem Tastendruck bis zu sechs mögliche Reaktionen und damit Funktionen ergeben, so ist das durchaus nicht so ganz ohne. Jeder der noch nie (länger) an einem Spectrum gearbeitet hat, wird nicht selten durch die mühsam zu erlernenden Tastaturmodi auf eine harte Probe gestellt. Wenn man mal weiß wie es geht, ist man allerdings dann sehr schnell und effektiv. Die BASIC Befehle sind im Speicherbereich $**0095** bis $**0204** zu finden. Analog dem Microsoft BASIC ist das letzte Zeichen eines Namens mit $80 versehen. Dies dient dazu, das Wortende einwandfrei zu erkennen. Im ROM nachfolgend ist die Zuordnung der Tastaturmatrix zu den ASCII Zahlenwerten realisiert. Durch die Auswertung im Rahmen der Syntaxprüfung wird dann erkannt, ob es sich um ein Zeichen oder ein BASIC Token handelt. Diese Tokens der entsprechenden Funktionen, die also 1 Byte Kürzel darstellen, sind im Bereich $**022C** bis $**028D** in Gruppen abgelegt.

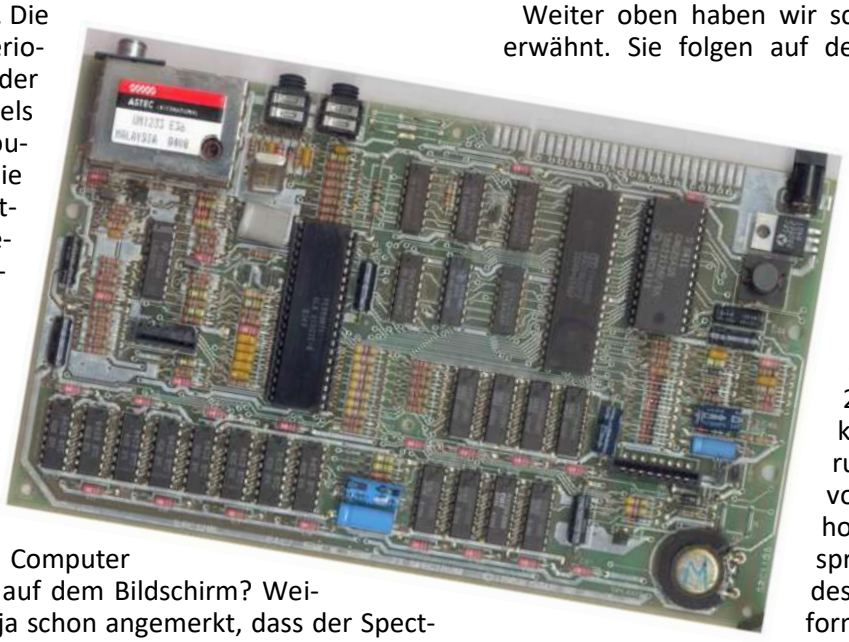
Das wesentliche Speichermedium in der Standardausführung des Spectrum bildete - seinerzeit üblich - der Kassettenrecorder. Die umfangreichen Routinen, die die Steuerung dieses Speichermediums ermöglichen, sind im Bereich von $**04C2** bis $**09F3** abgelegt. Beim ZX 81 gab es ja die Besonderheit, dass der Anschluss des Videosignals und der des Recorder-Ausganges identisch waren.

Hier hat sich beim Spectrum eine Änderung ergeben. In ähnlicher Weise sind beim Spectrum der Recorderausgang und die Farbausgabe des Bildschirmrandes verknüpft worden. Geübten Nutzern sind somit die temporären Zustände beim Arbeiten mit dem Speichermedium leichter erkennbar. Auch technische Fehler sind manchmal an den Streifen des Bildschirmrandes entzifferbar und können evtl. zur Fehlersuche bei Speicherproblemen herangezogen werden. Dazu, und das darf ich aus eigener Erfahrung sagen, muss man sich aber mit der Materie schon sehr gut auskennen.

Wesentliche Änderungen an der Recorder-Routine haben sich gegenüber dem ZX 81 hinsichtlich des Headers (Vorspann) ergeben. Zusätzlich wurde die Sicherheit beim Arbeiten mit dem Recorder gegenüber dem Vormodell doch wesentlich verbessert. Insbesondere die stabile Bildschirmausgabe beim Spectrum hilft hier doch sehr weiter. Neu hinzugekommen ist die **MERGE**-Routine, die das Anfügen von Programmteilen und Variablen erlaubt. Soviel ganz kurz zu diesem Thema.

Widmen wir uns jetzt den Soundroutinen und dem Tongenerator. Im Speicherbereich **$0385** bis **$04A9** stehen einige Routinen zu diesem Thema. Diese dienen der Steuerung des eingebauten Lautsprechers (auch wenn er diesen Namen nun wirklich nicht verdient). Es ist eben vielmehr ein Beeper, wie der englische Name schon richtig sagt. Der Anschluss selbst ist mehr als einfach gestaltet. Der Beeper hängt am Ausgangsport **$FE Bit 4**. Wenn ein Programm oder ein Befehl dort eine Null ausgibt, ist der Stromfluss eingeschaltet, eine eins schaltet ihn dann wieder aus. Wie inzwischen wohl fast jedem bekannt ist, besitzt der Spectrum im Gegensatz zu manch anderen 8-Bit-Computern keinen Soundcontroller. Deshalb muss das beim Spectrum der Prozessor übernehmen und durch entsprechende Ein- und Ausschaltimpulse im richtigen Rhythmus einen Sound erzeugen. Da es nicht ganz einfach ist mit dieser Methode einen einigermaßen reinen Ton zu erzeugen,

sind die Zeiten des Ein- und Ausschalten exakt einzuhalten. Damit kann man sich vielleicht vorstellen was passiert, wenn die Tonerzeugung beispielsweise durch einen Programminterrupt unterbrochen wird. Deshalb müssen während der Tonerzeugung die Interrupts notwendigerweise gesperrt werden. Das bedeutet aber auch, dass z. B. die Tastatur kurzzeitig nicht angesprochen werden kann. Die Werte der Periodendauer erhält der Spectrum mittels internen Rechenroutinen, indem die gewünschte Zeitdauer und die Frequenz in Periodendauer bzw. Zählimpulse und Anzahl der Perioden umgerechnet werden. Ihr seht also, es geht auch gerne etwas kompliziert.



Was wäre ein Computer ohne die Ausgabe auf dem Bildschirm? Weiter oben habe ich ja schon angemerkt, dass der Spectrum einen festen Bildschirmspeicher sein eigen nennt der pro Bildpunkt ein Bit reserviert. Dieser Bildschirmspeicher beginnt bei der Speicherstelle **$4000** (16384 – kommt bekannt vor?) und geht bis **$5800** (22528). Das sind dann die berühmten 6144 Bits eines Screens ohne Attributwerte. Das entspricht dann 192 Zeilen zu je 256 Punkten, vorausgesetzt man berücksichtigt, dass in ein Byte 8 Punkte passen. Wer aufgepasst hat, dem ist vielleicht aufgefallen, dass 192 Zeilen aber 24 Textzeilen entsprechen? Das rührt daher, dass der Spectrum einen aufgeteilten Bildschirmbereich besitzt. Nämlich die bekannten 22 Zeilen des oberen und die 2 Zeilen des unteren Bereiches. Der Benutzer kann innerhalb eines BASIC Programmes unter normalen Umständen nur auf die 22 oberen Zeilen zugreifen. Das sind dann wieder die bereits weiter oben erwähnten 176 vertikalen Bildpunktpositionen. Die unteren zwei Textzeilen werden für Eingaben und Systemmeldungen verwandt. Der untere Bereich kann sich jedoch dynamisch verändern, wenn entsprechend viel Text eingegeben wird. Das bedeutet dann aber auch, dass BASIC-Zeilen über die normale Breite einer Zeile hinauswachsen können. Beim Spectrum hätte man sich vielleicht auch mal einen besseren Editor, etwa so wie beim Commodore oder Epson-Rechnern gewünscht. Wer den guten alten Apple II dagegen kennt, ist mit den Editiermöglichkeiten des Spectrum dann ganz schnell wieder zufrieden. Eine vorher ausgewählte Zeile (siehe Cursorzeichen **>**) wird mittels des EDIT-Befehles zum Bearbeiten bereitgestellt. Man kann nun darin an jeder beliebigen Stelle Eingaben tätigen, doch werden vorhandene Zeichen nicht überschrieben oder gelöscht, sondern vielmehr weitergeschoben. Das ist manchmal ganz hilfreich, wenn man nicht gleich alles Vorhandene sofort wieder verlieren will. Andererseits muss man überflüssiges noch mit DELETE entfernen, ehe die Zeile dem Interpreter zur Syntaxprüfung übergeben werden kann. Diese Editierfunktion stammt noch vom ZX 81. Der Bildschirmspeicher ist ja bekannterweise pixelorientiert, d. h. alles was zu sehen ist, ist aus einzelnen Punkten dargestellt. Damit besteht die Möglichkeit, Zeichen über Zeichen darzustellen oder Linien durch Texte zu zeichnen. Alles kein Problem. grafische Figuren und Texte sind für den Spectrum einfach identisch.

Weiter oben haben wir schon kurz die Attribute erwähnt. Sie folgen auf den Speicherbereich der Punkte und haben eine Länge von 768 Bytes. Das bedeutet, dass für jedes Zeichenfeld in der Größe von 8*8 Bit eine entsprechende Farbkennzeichnung vorhanden ist. Das sind dann für den ganzen Bildschirm 24*32 Felder. Damit kann man beim Spectrum also nicht wirklich von der Ausgabe einer hochauflösenden Grafik sprechen. Das ja auch deshalb, weil sich Farbinformationen nur pro Zeichenfeld und nicht pro Zeichen festlegen lassen.

Ein kleines Rechenexempel dazu? Wenn wir wirklich acht Farben pro Bildschirmpunkt darstellen wollten, bräuchten wir 3 Bit pro Punkt. Somit bräuchten wir 3*6144 Byte = 18432 Byte. Damit wäre der Bildschirmspeicher dreimal so groß wie jetzt und wir hätten kaum mehr Platz für das Programm. Wenn man sich aber überlegt, dass auch der derzeitige Weg schon fast 7 KByte nur für den Bildschirm benötigt, kommt man von solchen Forderungen schnell wieder ab. Während beim ZX 81 die Bilderzeugung noch mittels ausgeklügelter Software (Interrupt gesteuerter Zeilenaufbau!!) erfolgt, bedient sich der Spectrum hier der Hardware. Das geht nur, da der Spectrum über eine wesentlich höhere Taktung und damit Rechenleistung verfügt. Die Bilderzeugung belastet den Z 80 nur unmerklich, etwas mehr kostet ihn die Verwaltung des Bildschirminhaltes.

Neben dem Bildschirm ist ja wohl der Drucker das am weitesten verbreitete Medium zur Ausgabe. Eine Speicherseite von 256 Bytes im Bereich **$5b00** (23296) dient der Sammlung der Druckerinformationen. Beim Spectrum ist auch der Druckerspeicher punktorientiert und deshalb sind auch hier Grafik und Text ohne weiteres mischbar. Farbinformationen benötigt der Drucker keine. Ich werde später nochmals an anderer Stelle in einigen Details zu Fragen des Druckens zurückkommen.

Was begegnet dem geneigten User aber als erstes nach dem »Hochfahren« des Systems? Gemeint ist nicht die Statusmeldung mit der sich der Spectrum als einsatzbereit meldet, nein wir sprechen von dem an dieser Stelle bereits aktiven BASIC Interpreter, der auf die die Befehlseingabe des Users wartet. Der im ROM integrierte Interpreter leistet die Aufgaben der Zeileninterpretation, der direkten Befehlsausführung und der Berechnung mathematischer Ausdrücke, wozu er die arithmetischen Routinen und den vorhandenen Gleit-

kommarechner enthält, der z. B. Polynomberechnungen der trigonometrischen Funktionen ausführt. Der Spectrum besitzt - im Gegensatz zum ZX 81 - einen erweiterten Umfang dieser Funktionen nebst deren Inversen. Der Befehlsvorrat wurde gegenüber dem ZX 81 merklich erweitert. Ein weiter bestehendes Problem sind logische Befehle wie z. B. **AND** und **OR**. Sie sind für fast alle Interpreter nicht einfach darstellbar, da sie grundsätzlich anders reagieren und sich auch keineswegs für logische Prüfungen innerhalb eines Bytes oder innerhalb dezimaler Zahlen bis 255 eignen.

Würde man hier auch noch die Behandlung von Bits innerhalb von Bytes berücksichtigen, müsste dieses durch ein enorm aufwendiges Teilen und Prüfen geschehen. Spectrum-typisch sind ja die bekannten Grafikbefehle **PLOT**, **DRAW** und **CIRCLE**. Wie wohl sicherlich bekannt ist, verwendet **CIRCLE** beim Zeichnen der Kreisbögen sehr kleine Geradenstücke. Dies geschieht vor allem aus Gründen der Performance.

Wohl jeder, der noch niemals mit einem Sinclair Rechner und hier insbesondere mit dem Spectrum gearbeitet hat, wird an der manchmal schon sehr eigenwillig zu nennenden bis zu sechsfachen Tastenbelegung schier verzweifeln. Ich habe das ja weiter oben schon erwähnt. Zugegeben, es braucht seine Zeit, bis man sich an die einzelnen Modi gewöhnt hat, aber wenn man sie einmal intus hat, will man sie auch nicht mehr missen. Hintergrund war seinerzeit, dass man unbedingt erreichen wollte, alle BASIC Befehle über einen so genannten *»Ein-Tasten-Druck«* abzubilden.

Bedenkt man die verschiedenen Shift-Modi, die durch die zu drückende Taste zu bewältigen sind, erhöht sich durch das Drücken der verschiedenen Tastenkombinationen, um in den entsprechenden Modus zu gelangen, nicht selten die Anzahl der zu drückenden Tasten über die Buchstaben hinaus, die der endgültig zu erreichende Befehl dann wirklich hat. Ich hoffe der Satz war jetzt nicht zu verschlungen und ihr konntet verstehen, was ich damit sagen wollte. Als Beispiel für die Kuriosität dieses Verfahrens seien hier die roten Token unter den Buchstabentasten aufgeführt (im ROM ab der Adresse **$0249** zu finden). Um die eckigen oder geschweiften Klammern, also ein einziges Zeichen zu erreichen, muss man zuerst **Symbol Shift** und **Caps Shift** drücken, um in den **E**-Modus zu gelangen. Weiter geht es dann mit der Taste **Symbol Shift** zusammen mit der Taste für das Zeichen, das man haben wollte. Zugegebenermaßen ist das ein wenig grotesk.

Weiter im ROM kommt man in den letzten drei Speicherseiten, also von **$3D00** bis **$3FFF**, zu den darstellbaren Zeichen (Text und Grafik). Wie weiter oben schon gesagt, bestehen die Zeichen ja aus einer 8*8 Matrix und belegen deshalb auch entsprechend viel Platz. Wer nachgezählt hat wird feststellen, dass der Spectrum insgesamt 96 Zeichen »kennt«. Angemerkt sei an dieser Stelle, dass die groben Grafikzeichen im »Vierteldesign« nicht im ROM stehen, sondern berechnet werden. Ungewöhnlich am Zeichensatz ist, dass die Zeichen an allen Enden oder besser gesagt an den Kanten eine Punktposition freilassen. Das ist bei anderen Rechnern so nicht zu finden. Beim Andruck der Zeichen auf dem Bildschirm besitzen die Zeichen ja keinen Abstand sondern grenzen direkt aneinander. Aus diesem Grund, gehen die Zeichen eben in der Matrix nicht ganz bis zum Rand, da sie sonst auf dem Bildschirm ineinander laufen würden. Diese Maßnahme trägt aber in der Folge dann nicht unbedingt zur Klarheit des Zeichensatzes bei, denn durch die besagte Maßnahme reduziert sich natürlich dann auch die Matrix je Zeichen von 8*8 auf 6*6 Pixel. Deshalb ist der Zeichensatz des Spectrum ja auch nicht unbedingt als sehr präzise und gut lesbar bekannt.

Wie bereits weiter oben angedeutet, wollen wir uns jetzt nochmal kurz mit der Thematik von anzuschließenden Peripheriegeräten und den daraus resultierenden I/O-Problemen befassen. Allerdings sei hier nur die Softwareseite betrachtet. Alles andere ist Sache der Hardwarespezialisten, zu denen ich mich bestimmt nicht zählen kann. Auf der Tastatur sind bekannter Weise auch Befehle für ein rudimentäres Speichersystem in Form von externen »Floppies« zu finden. Die Befehle **CAT**, **ERASE**, **FORMAT** und **MOVE** laufen im ROM an der Speicherstelle **$1793** schlicht und einfach auf ein „**INVALID STREAM**“. Die Steuersoftware ist im Spectrum nicht vorhanden und muss durch das externe Speichergerät selbst (im eigenen ROM) *»mitgebracht«* werden. Dazu gab es bereits viele interessante Artikel, insbesondere über die erhältlichen Diskettensysteme und deren Handhabung. An dieser Stelle sei auf die entsprechenden Fachbeiträge hier in unserem Clubmagazin verwiesen (wer mehr wissen will, was es spezielle zu den einzelnen Themen dazu gibt, sei auf den »Index« der Clubartikel oder die Internetsuche verwiesen). Im Netz findet man ja *fast* alles.

Für die Ansprache von Input- und Output-Erweiterungen besitzt der Spectrum die Befehle **IN** und **OUT**. Diese sind in ihrer Art völlig identisch mit dem Z80 Code (nachzulesen u. a. bei R. Zaks). Die Ausgabe erfolgt bei Speicherstelle **$1E7D** in Form von **OUT (C),A**, die Eingabe bei Speicherstelle **$34A8** mit **IN A,(C)**. Wer sich näher mit dem Thema auseinander setzen will, dem sei die Lektüre des Handbuches bzw. die Literatur rund um den Z80 Code empfohlen.

Der Spectrum benutzt ja für seine einzelnen Systemeinheiten wie die Tastatur, den Beeper, Recorderanschluss oder Drucker ganz spezielle Ports. Wenn der Anwender diese kennt, kann er auch aus BASIC heraus eigene externe Geräte betreiben.

Liebe Mituser, damit möchte ich den kleinen Rückblick unter die »Gummimatte« dann auch schon wieder beschließen. Wenn man schon viele Jahre mit dem Spectrum gearbeitet hat, ist es immer wieder mal interessant zu dessen Wurzeln zurückzukehren und sich so manches ins Gedächtnis zu rufen.

Wie wir gesehen haben, ist der Spectrum schon ein merkwürdiger Geselle, besonders wenn man die beschriebenen Besonderheiten betrachtet. Aber vielleicht ist das auch der Grund, warum die Arbeit mit ihm so viel Spaß macht. Für unentdeckte Fehler darf ich mich hier an dieser Stelle gleich entschuldigen - sie waren keine Absicht.

© Harald Lack

»WILDE PIXEL«
WIE DAS BILD AUF DEN BILDSCHIRM KOMMT
VON CHRISTOF ODENTHAL

Eigentlich wollte ich in diesem Artikel hauptsächlich über den ORIC von *Tangerine* und seinen „verrückten“ Bildaufbau schreiben, aber jetzt ist doch ein größerer Rundumschlag draus geworden!

Auf die Geschichte des *ORIC* werde ich hier nicht weiter eingehen, das kann man in der SUC-Session Nr. 227 nachlesen. Nur so viel: Der *ORIC* war als Konkurrent für den ZX Spectrum gedacht, konnte diesen Status aber nie erreichen. Ich möchte aber anmerken dass ich den *ORIC Atmos* mit seinem schwarz-roten Gehäusedesign und seiner »richtigen« Tastatur für einen der schönsten (wenn auch nicht besten) 8-Bit Computer der Ära halte.

Der Hauptgrund für diesen Artikel (in dem es vorwiegend um die Grafikdarstellung geht) war ebendiese Verwandtschaft und der trickreiche Aufbau des Bildspeichers, der bei etwa gleichen Voraussetzungen eine größere Zahl an unterschiedlichen Farben in einem kleineren Bereich ermöglicht – allerdings muss man dafür andere Einschränkungen in Kauf nehmen.

Durch meine Sammlung an Homecomputern kenne ich viele Systeme und ihren Bildspeicher-Aufbau, vor allem wenn es um hochauflösende Grafik geht. Grund für die teilweise komplexen Aufbauten sind vor allem Speicherersparnis (RAM-Speicher war damals sehr teuer) und technische Gründe, d.h. eine Vereinfachung der Hardware und damit ebenfalls eine Kostenreduktion. Die meisten Systeme müssen auch mit einer Bremse des Prozessors leben während der Grafikchip auf den Bildspeicher zugreift. Manchmal betrifft das nur bestimmte Speicherbereiche, wenn der Grafik-Chip nicht auf den gesamten Speicher zugreifen kann – das wird dann meist *»Contention Memory«* oder *»Chip-RAM«* (im Gegensatz zum *»Fast-RAM«)* genannt. Beim *MSX 1* gibt es keine Bremse da der Videochip seinen eigenen Speicher besitzt – allerdings ist er nicht per direktem Speicherzugriff sondern nur per *OUT*-Befehl zu erreichen, was den Zugriff wieder ausbremst!

SYSTEME MIT GETRENNTEM GRAFIK- UND FARBSPEICHER

Bei diesen Systemen (z.B. *ZX Spectrum*, *C64*, *MSX 1* Mode 2) ist die Grafik monochrom (jedes Bit ein Pixel) und die Farbe wird über den separaten Farbspeicher bestimmt (über sogenannte „Farb-Attribute“) – meist für einen Bereich von 8x8 (ZX *Spectrum*, *C64*) oder 8x1 (*MSX 1*) Pixeln.

SYSTEME MIT GEMEINSAMEN GRAFIK- UND FARBSPEICHER

Bei diesen Systemen (z.B. *Amstrad/Schneider CPC*, *Atari*, *MSX 1* Mode 3) bestimmen mehrere Bits eines Bytes die Farbe für das entsprechende Pixel.

Die Farbpalette selbst ist dabei entweder fest (*ZX Spectrum*, *C64*, *MSX 1*, *ORIC*) oder kann aus einem größeren Farbraum gewählt werden (*CPC*, *Atari*).

MISCHFORMEN UND VARIANTEN

Beim *C64* Hires Multicolor-Modus werden zwei Bits eines Byte zusammengefasst und verweisen entweder auf einen von drei Farbattribut-Speichern oder auf ein Register mit einer gemeinsamen Hintergrundfarbe. In einem 8x8 Pixel-Bereich können so vier verschiedene Farben dargestellt werden, dafür ist die Auflösung um die Hälfte reduziert.

Beim *ORIC* können sich Bytes mit Monochrom-Grafik und Farbattribute im selben Speicher abwechseln, d.h. die Attribute sind nicht, wie zum Beispiel beim *C64* oder *ZX Spectrum*, in einem separaten Speicher.

Beim *MSX 1* möchte ich noch den Modus 1 erwähnen, der zwar ein Text-Modus (mit frei definierbarem Zeichensatz) ist, aber eine ungewöhnliche Farbgebung besitzt: Im Zeichensatz können jeweils acht aufeinanderfolgenden Zeichen eine gemeinsame Farbe zugewiesen werden, d.h. die Zeichen mit Code 0-7 können eine Farbe haben, die Zeichen mit Code 8-15 eine andere usw.

AUFBAU DES GRAFIKSPEICHERS

Daneben gibt es auch Unterschiede im Aufbau des Grafikspeichers. Man denkt die Pixel, die man auf dem Bildschirm sieht, sind (in Form von Bytes) genauso im Speicher abgelegt, d.h. zeilenweise von links oben nach rechts unten. Dem ist aber nicht so! Das kann man beobachten wenn man den Bildspeicher linear mit Bytes füllt und dabei den Bildschirm betrachtet – Euch ist das sicher schon mal beim Laden eines Screens aufgefallen! Folgendes Programm (auf dem *ZX Spectrum*) verdeutlicht es ebenso – langsam zum Zuschauen:

```
5 CLS
10 FOR n=16384 TO 22527
20 POKE n,255
30 NEXT n
```

Wenn ich im Folgenden von Zeilen rede, meine ich damit Pixel-Zeilen, nicht Charakter-Zeilen – eine Charakter-Zeile sind meist acht Pixel-Zeilen!

Es gibt u.a. folgende Methoden (auch Kombinationen daraus):

- Einfachster Fall: Die Bytes mit den Pixeln werden zeilenweise, Zeile für Zeile, gelesen und angezeigt (Atari).
- Zeilensprungverfahren: Dabei wird ebenfalls links oben in Zeile 1 angefangen. Die Bytes/Pixel folgen dann direkt nebeneinander bis das Zeilenende erreicht ist. Als nächstes folgt dann aber nicht Zeile 2 sondern es werden ein paar Zeilen übersprungen (beim *ZX Spectrum* und *CPC* hat eine Zeile acht Pixel Abstand zur nächsten). Am Ende des Bildschirms (oder Blocks, siehe unten) folgt erst Zeile 2 usw.
- Beim *C64* und *MSX 1* wird auch links oben angefangen und erst einmal ein Bereich von der Größe eines Charakters, d.h. 8x8 Pixel bzw. acht Bytes, gefüllt – das sind dann 8 Zeilen untereinander mit der Breite von einem Byte. Dann folgen, wieder angefangen auf der Höhe der ersten Zeile, die nächsten 8x8 Pixel zur Rechten. Am Bildrand angekommen wird wieder links acht Pixel-Zeilen tiefer fortgesetzt.
- Der Bildschirm kann manchmal zusätzlich in Blöcke unterteilt sein – Ein Block geht dabei vom linken bis zum rechten Bildrand, füllt aber nach unten hin nicht den ganzen Bildschirm, sondern nur einen Teil davon (beim *ZX Spectrum* und *MSX 1* Mode 2 sind es drei vertikale Blöcke). Beim Füllen des Bildspeichers wird ein Block nach dem anderen gefüllt, nach einem der obigen Verfahren, bevor der nächste Block an der Reihe ist.

Der *Amiga* macht es übrigens ganz anders, er hat sogenannte »Bitplanes« – das sind im Prinzip separate Grafikspeicher, die übereinander gelegt werden. Jede zusätzliche Bitplane fügt ein Farb-Bit hinzu, d.h. mit nur einer Bitplane ist die Grafik monochrom, mit zwei Bitplanes kann man schon vier Farben darstellen, mit drei acht usw. Nachteil ist dass man so immer mehrere verschiedene Speicher gleichzeitig adressieren / beschreiben muss um eine mehrfarbige Grafik auf den Bildschirm zu kriegen. Um das auszugleichen hat der Amiga zwar einen Spezialchip namens »Blitter« um große Datenmengen zu bewegen, trotzdem ist es ein Nachteil z.B. wenn schnelle Grafik für 3D-Ego-Shooter benötigt wird. Der *Amiga* CD32 hat übrigens einen weiteren (exklusiven) Spezialchip namens *»Akiko«* der die *»Chunky to Planar«* Konvertierung beherrscht (eine Verteilung von Farb-Bits aus einem gemeinsamen Speicher *»chunky«* auf mehrere Bitplanes *»planar«*) – der Chip war aber hauptsächlich zur schnellen Darstellung von Videos von CD gedacht und wird selten genutzt.

Der große Nachteil dieser teilweise recht komplizierten Bildspeicher ist, dass man ebenso aufwendige Algorithmen braucht um einen rechteckigen Bereich hineinzuschreiben, z.B. einen Sprite oder Hintergrundgrafik – und das macht die Programme langsamer!

Die grafischen Fähigkeiten des ORIC

Nach dem obigen Ausflug jetzt aber zum *ORIC*! Er kann 8 verschiedene Farben darstellen, besitzt einen Text-Modus (»TEXT«, »LORES 0« oder »LORES 1«) mit 40x28 Zeichen (wobei ein Zeichen aus 6x8 Pixeln besteht) und einen Grafik-Modus (»HIRES«) mit bis zu 240x224 Pixeln. Das Besondere ist, dass eine Zeile auf dem Bildschirm zwar aus 40 Bytes besteht – was eine Auflösung von 40 Byte x 8 Bits pro Byte = 320 Pixeln vermuten ließe – sich die Auflösung aber durch die Benutzung von nur sechs Bits pro Byte auf 240 Pixel reduziert. Die restlichen zwei Bits werden zur Steuerung der Bildausgabe benutzt. Der Unterschied zwischen den beiden Modi ist, dass der Bildspeicher im Text-Modus auf Zeichen aus dem Zeichensatz verweist, während er im Grafik-Modus direkt die Pixel enthält. Der Text-Modus belegt 3040 Bytes Speicher (inklusive Zeichensatz) und der Grafik-Modus 10210 Bytes (mit drei Textzeilen und Zeichensatz) – der reine Grafikspeicher wäre 8960 Bytes. Im Vergleich dazu: Der Bildspeicher vom *ZX Spectrum* ist 6912 Bytes groß.

Die Attribute des ORIC

Wie oben bereits erwähnt hat der *ORIC* keinen separaten Attribut-Speicher. Grafik und Attribute teilen sich denselben Speicher – das heißt aber auch dass an einer Stelle, an der ein Attribut ist, kein Charakter und keine Grafik sein kann! Wenn man also die Farbe für die folgenden Zeichen oder Pixel ändern will, muss man damit leben dass eine sechs Pixel breite Lücke in Vorder- oder Hintergrundfarbe entsteht.

Die Attribute werden »serielle Attribute« genannt weil ihr Effekt spätestens am Zeilenende endet (mit Ausnahme der Video-Kontroll-Attribute – die bleiben bis sie wieder geändert werden). Bei jedem Start einer neuen Zeile wird die Hintergrundfarbe wieder auf schwarz und die Vordergrundfarbe auf weiß zurückgesetzt. Ein Attribut wird dadurch markiert dass die Bits 6 und 7 null sind – ein Byte im Bildspeicher mit einem Wert von 0-31 = Bits 1-5 (bzw. 128-159 wenn Bit 8 auch gesetzt ist) stellt also ein Attribut dar. Werte zwischen 32-127 sind je nach Modus entweder ein Charakter oder Pixel (Bit 1-6 = Pixel, das 7. Bit muss dabei gesetzt sein). Bit 8 invertiert die Farbe des aktuellen Zeichens, der sechs Pixel oder des aktuellen Attributs. Der ORIC besitzt leider kein »Bright«-Attribut, d.h. seine acht Farben gibt es nur in einer Helligkeit. Auch der Border ist immer schwarz und kann nicht geändert werden.

Attribute zur Video-Steuerung

Es gibt aber nicht nur Attribute um die Vorder- oder Hintergrundfarbe zu ändern, man kann sie auch blinken lassen (wie beim *ZX Spectrum* – wobei im Gegensatz dazu nur die Vordergrundfarbe blinkt, nicht die Hintergrundfarbe). Allerdings kann man mit den Attributen auch im Text-Modus zwischen zwei Zeichensätzen (je 128 Zeichen) wechseln oder die Anzeige der Zeichen auf doppelte Höhe (d.h. 6x16 Pixel) umschalten – in jeder geraden Zeile ist dann die obere Hälfte des Zeichens zu sehen und in jeder ungeraden die untere Hälfte!

Selbst der Text- und Grafik-Modus wird über die Attribute gesteuert – und das entweder mit 50 oder 60 Hertz Bildwiederholfrequenz! So kann man Grafik und Text mischen, sollte sich dabei aber für eine der beiden Frequenzen entscheiden, sonst könnte ein Videosignal entstehen dass der Fernseher / Monitor nicht darstellen kann!

Attribut-Wert	Vordergrund-Farbe	Attribut-Wert	Hintergrund-Farbe
0	Schwarz	16	Schwarz
1	Rot	17	Rot
2	Grün	18	Grün
3	Gelb	19	Gelb
4	Blau	20	Blau
5	Magenta	21	Magenta
6	Cyan	22	Cyan
7	Weiß	23	Weiß
Attribut-Wert	**Effekt**	**Attribut-Wert**	**Video-Modus**
8	Standard Zeichensatz	24	Text Modus 60 Hz
9	Alternativer Zeichensatz	25	
10	Doppelt Standard	26	Hires Modus 60 Hz
11	Doppelt Alternativ	27	
12	Blinkend Standard	28	Text Modus 60 Hz
13	Blinkend Alternativ	29	
14	Doppelt blinkend Std.	30	Hires Modus 60 Hz
15	Doppelt blinkend Alt.	31	

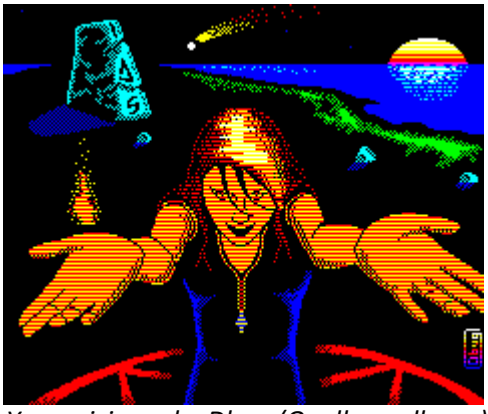
Tabelle : Attribut-Werte

EIN PAAR TRICKS

Ebenso wie beim *ZX Spectrum* gibt es ein paar Tricks um die Limitationen der Grafik ein wenig zu verstecken. An jedem neuen Zeilenanfang werden die Farben ja (wie oben beschrieben) immer wieder automatisch auf schwarz (Hintergrund) und weiß (Vordergrund) zurückgesetzt – üblicherweise setzt man dann dort seine eigenen Farbattribute für die folgende Grafik, wofür man zwei Byte braucht (je eins für Hintergrund und Vordergrund), dadurch gehen 2x6 = 12 Pixel verloren. Da der Border immer schwarz ist, besteht ein Trick darin im ersten Byte der Zeile zuerst das Attribut für die Vordergrundfarbe zu setzen und erst im zweiten Byte das für die Hintergrundfarbe – so wirkt es als ob der Border um sechs Pixel breiter ist (weil der Hintergrund für diese ersten sechs Pixel dann immer noch schwarz ist) und man verliert scheinbar nur sechs Pixel (im zweiten Byte) für die man keine Grafik setzen kann, weil da das Vordergrundfarbattribut gesetzt wird. Man kann aber auch einfach gar kein Attribut setzen und die Farben damit einfach schwarz / weiß lassen um die vollen 240x224 Pixel für ein Monochrom-Bild zu nutzen.

»Beast«, by Twilighte (Quelle: osdk.org)

Beim *ZX Spectrum* kann man (ohne Tricks) nur zwei Farben für jeden 8x8 Pixel-Bereich wählen. Beim *ORIC* geht das für jeden 6x1 Pixel-Bereich – man kann also in jeder Pixelzeile neue Farben setzen. Wenn man jetzt in jeder zweiten Zeile die Farben wechselt, verschwimmen die Farben aus größerer Entfernung zum Bildschirm zu einer neuen.



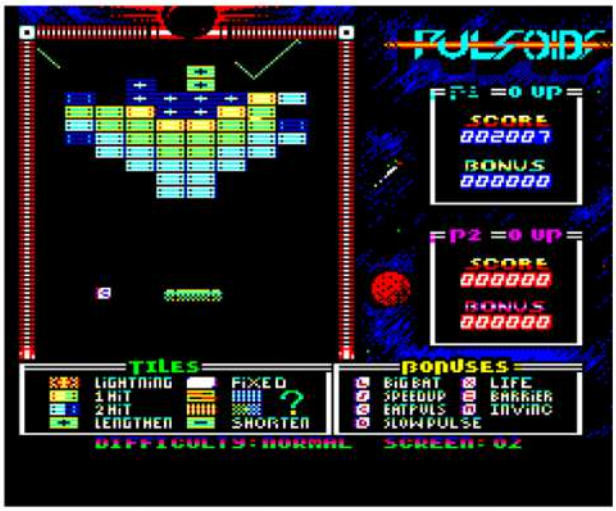
»Yessagician«, by Dbug (Quelle: osdk.org)

Ein weiterer Trick besteht in dem geschickten Einsatz des Farbinvertierungs-Bits (Bit 8, Wert 128) – das kann in jedem Attribut, Charakter oder Grafikbyte verwendet werden und wechselt sofort die Farben (anhand folgender Tabelle) ohne dass ein Farbattribut verwendet und damit 6 Pixel verschenkt werden müsste.

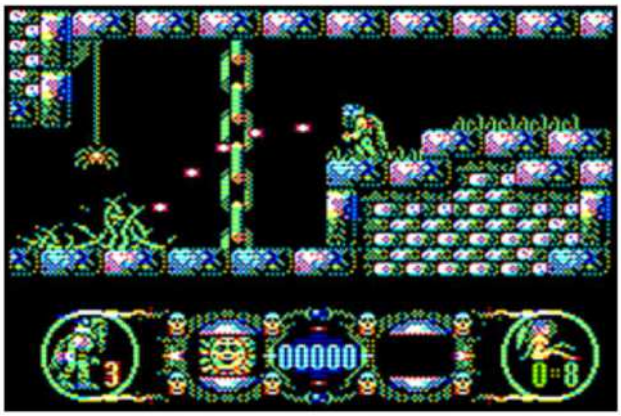
Originalfarbe	Invertierte Farbe
Schwarz	Weiß
Rot	Cyan
Grün	Magenta
Gelb	Blau
Blau	Gelb
Magenta	Grün
Cyan	Rot
Weiss	Schwarz

Tabelle : Invertierte Farbwerte

»Ghost Gobbler«, (Quelle: osdk.org)



»Pulsoids«, by Twilighte (Quelle: osdk.org



»Stormlord«, by Twilighte (Quelle: osdk.org)

Die meisten Spiele für den *ORIC* werden übrigens (genauso wie beim *C64*) im Text-Modus programmiert. Das ermöglicht schnelle Animationen – einfach durch ändern eines Zeichens im Zeichensatz, weil sich dann schlagartig alle Stellen auf dem Bildschirm ändern, an denen dieses Zeichen verwendet wird – und das ohne zeitaufwendiges Kopieren durch den Prozessor – das erledigt der Grafikchip sozusagen zum Nulltarif!

Zuletzt möchte ich einen weiteren großer Nachteil des *ORIC* (neben einem fehlenden BRIGHT-Attribut) nicht verschweigen: Es gibt keinen Interrupt (Signalisierung), wenn ein neues Videobild aufgebaut wird – nur damit kann man z.B. animierte Grafik mit dem Bildaufbau synchronisieren um ein Flackern zu vermeiden. Es gibt aber eine Hardware-Modifikation mit dem Kassetten-Interface um dieses Signal nachzurüsten. Das übliche Problem ist dabei halt, dass leider nicht jeder diesen Umbau hat, so dass man nicht davon ausgehen kann dass es vorhanden ist.

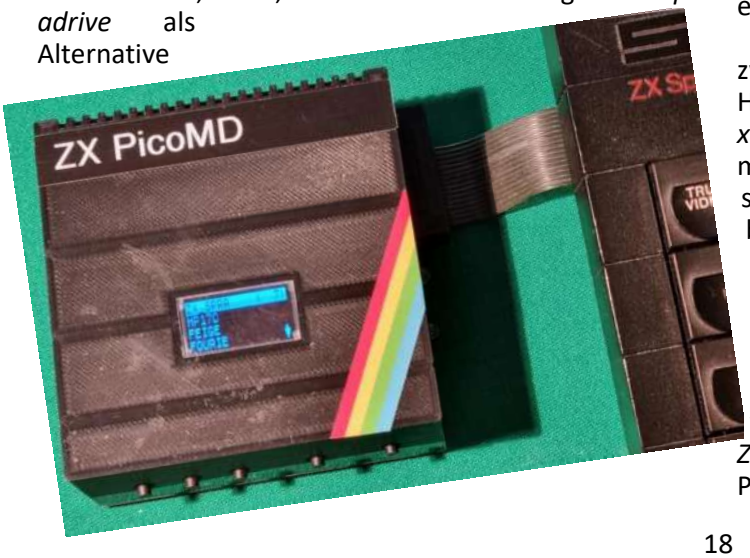
»DAS ZXPICOMD«

EINE WEITERE ALTERNATIVE ZUM MICRODRIVE/IF1

VON HEINZ JUNEK, 08. JANUAR 2023

Noch eine Alternative zum Microdrive am IF1: Das ZXPicoMDine moderne Alternative zum unzuverlässigen Microdrive mit allen dessen Vorzügen und mehr.

In Heft 233, S. 18, hatte ich einen Beitrag zum *Oqtadrive* als Alternative zum *Microdrive* am *IF1* mit klassischen »IF1-feeling« aber 100%er Zuverlässigkeit geschrieben. Dieses Gerät ist gemeinsam von *xelalex* und *TomD* entwickelt worden. Parallel dazu hat auch in `https://github.com/TomDDG/ZXPicoMD` sein eigenes Produkt *»ZXPicoMD«* herausgebracht.



Technisch sind beide Geräte bzgl. der Kommunikation zwischen dem Gerät und dem IF1 identisch. Der Hauptunterschied beider Geräte liegt m.E. darin, dass *xelalex* vor allem Wert auf die WLAN Kommunikation mit dem PC setzt, während *TomD* mehr Wert auf eine selbständige „Daseinsweise“ des *ZXPicoMD* legt. Das bedeutet insbesondere, dass das Archiv der mdr-files beim *ZXPicoMD* weitgehend auf der SD-Karte verwaltet wird, während xelalex dies hauptsächlich auf dem PC macht.

Das *ZXPicoMD* verfügt über ein eigenes, kleines OLED-display., mit dem das Gerät auch unabhängig vom IF1 gesteuert werden kann. Die Bedienung des *ZXPicoMD* erfolgt (neben den IF1-Befehlen) mittels fünf Push-Buttons. Hier können verschiedene Funktionen

angewählt werden: Laden und Speichern von Cartridges aus dem Archive bzw. in das Archive, Anzeigen von Cartridge-Inhalten (s. Abbildung) und auch Formatieren. Echte Sinclair-Microdrives können zusätzlich angeschlossen werden und über die Push-Buttons an beliebiger Stelle in die Laufwerkskette eingefügt werden. Der Name »*ZXPicoMD*« rührt von der Verwendung des *Raspberry Pico* her. Da dieser wesentlich billiger als ein Raspberry *Pi Zero* ist und ein *Arduino nano* gar nicht gebraucht wird, sind die Baukosten wesentlich günstiger. Inklusive Gehäuse liegen sie bei etwa 20€. Durch die Verwendung des *Raspi Pico* ist das Gerät nach weniger als einer Sekunde betriebsbereit, und es braucht vor dem Abschalten des *ZX Spectrums* auch kein langsames Herunterfahren.-Diese beiden Prozesse dauern beim *Oqtadrive* ca. 30 Sekunden. Die Installation des *ZXPicoMD* erfodert auch keinen LINUX Zugang, was die Schwelle für einen Nachbau herabsetzt. Zukünftig ist ferner eine Kommunikation mit dem PC über WLAN vorgesehen. Beim Nachbau ist daher zu empfehlen, gleich einen »*Raspberry Pico W*« zu verwenden. Insgesamt macht dieses neue Gerät mindestens genauso viel Spaß wie das *Oqtadrive*.

»DIZZY 4 - MAGICLAND«

SPIELBESCHREIBUNG UND RÜCKBLICK VON HARALD LACK

R. Fred Williams, Peter J. Ranson, Chris Graham, David Whittaker und Neal Vincent
(1990 - Code Masters Ltd)
Verwendete Hardware: Original ZX Spectrum 48k

Hallo Freunde der *Dizzy* Spiele!!

In meinem heutigen Beitrag geht es diesmal um keine (Text-)Adventurelösung sondern die Lösungen zu klassischen *Dizzy* Spielen mit *Dizzy 4 - Magicland Dizzy*. Das Spiel kam 1990 auf den Markt und wurde von *Code Masters Ltd* vertrieben. Verantwortlich für die Programmierung waren hier R. *Fred Williams, Peter J. Ranson, Chris Graham, David Whittaker* und Neal *Vincent*. Somit war ab hier keiner der *Oliver Twins* mehr an Bord obwohl die Story natürlich auf deren Spielfigur basiert, was der Qualität meines Erachtens aber nicht abträglich war.

Das ganze lässt sich auf einem *Spectrum 48K* einwandfrei spielen lief aber auch auf den größeren Brüdern. Vom Spielablauf hat sich gegenüber den Vorgängern dieser Reihe nichts wesentliches geändert, es gibt wie immer noch eine Menge Rätsel zu lösen, man muss viel rumlaufen, mit den vorgefundenen Gegenständen an der richtigen Stelle hantieren und versuchen weiterzukommen. Dabei bekommt man so manchen Tipp, wenn man mit den anzutreffenden Charakteren ein »Schwätzchen« hält. Aber hier bitte auf die richtigen »Gesprächspartner« achten. Unsere Aufgabe in diesem Spiel ist es, unsere verzauberten Freunde aus der Macht dem bösen Zaks zu befreien und schließlich Zaks selbst zu besiegen. Aber damit noch nicht genug. Danach sollten wir versuchen, aus dem Magicland zu verschwinden und wieder in unsere vertraute Heimat zurückzukehren. Dies ist uns aber erst dann möglich, wenn wir mit dem Teufel ein Geschäft gemacht haben, denn nur dieser kann uns zurückteleportieren. Dafür verlangt er aber die Übergabe von 30 Diamanten, die wir vorher eingesammelt haben müssen. Noch kurz ein paar Anmerkungen dazu. Die Mehrzahl der Diamanten (genau genommen 27) sind direkt sichtbar und als Spieler neigt man dazu sie einfach einzusammeln. Da uns bei diesem Programm aber immer wieder Gefahren unterschiedlichster Art drohen (dazu im weiteren Verlauf noch näheres), die uns Energie kosten und wir ohnehin nur drei Leben zur Verfügung haben, sollten wir die Diamanten nur nach und nach - also sehr systematisch aufnehmen, denn sie bringen Energie zurück und können uns manchmal das Leben entscheidend verlängern. Natürlich dürfen wir dabei nicht vergessen auch am Ende alle eingesammelt zu haben. Wer oben aufgepasst hat wird bemerkt haben, dass da noch drei Diamanten fehlen; nun diese sind wieder hinter mancherlei Dingen versteckt und so natürlich schwer zu finden. Ich habe euch aber unten aufgelistet, wo alle Diamanten zu finden sind. Was welcher Gegenstand bewirkt und was man mit den Sachen machen muss, könnt ihr der nun folgenden Aufstellung entnehmen:

- Backdoor key → öffnet den Hintereingang zum Schloß
- A beg of rubbish → absolut nutzlos
- Empty milk bottle → absolut nutzlos
- The gold cross → hilft gegen Vampire
- Power pill → beseitigt die Geister über dem Wasser
- The handle → zieht den Eimer am Brunnen hoch
- The eatme cake → absolut nutzlos
- A poisoned apple → zur Hexe Glenda bringen
- A bucket → mit heißem Wasser aus dem Geysir füllen und dann zu Denzil bringen, um ihn aus dem Eis zu schmelzen
- A cloth duster → bei Dozy damit die Lampe reiben um Zaks zu rufen
- A big stick → damit die Ziege schlagen
- A black cat → zur Hexe Glenda bringen
- A dagger → damit die Ziege losschneiden und bei Dylan ein Blatt abschneiden
- A leaf → zur Hexe Glenda bringen
- The goat → mit dem Stock schlagen um den Troll von der Brücke zu stoßen
- Dylan → den Zaubertrank von Glenda bringen

- Denzil → mit dem Eimer heißen Wasser schmelzen - wir erhalten den personal stereo
- Prince charming → den Frosch Dora bringen
- Hexe Glenda → black cat, leaf, poisoned apple und torch bringen
- Dozy → lighning rod, lamp und duster bringen und dann Zaks rufen
- Troll → mit der Ziege vertreiben
- The Bard → den personal stereo bringen
- The drinkme potion → zu Daisy bringen um sie zu schrumpfen
- Dora the frog → zu Prince charming bringen
- A personal stereo → zum Bard bringen - wir erhalten die magic pipes
- Some magic pipes → zur Ratte im Brunnen bringen und ihr den Marsch blasen
- The Rat → mit den magic pipes vertreiben
- Something sticky → damit das Schwert aus dem Fels ziehen
- An ancient lamp → mit dem Tuch reiben
- Ghosts → mit der Power pill vertreiben
- A lightning rod → zu Dylan bringen
- Lever → ermöglicht einen Weg über das Wasser
- A Railing → verdeckt Diamanten
- Vampir → mit dem Goldkreuz vertreiben
- Monkey → den Nüssen ausweichen
- Excalibur → damit wird man zum König und kann die Dame im Schach schlagen
- The torch → öffnet Eingang zu Daisy - danach zur Hexe Glenda bringen
- Daisy v ihr die drinkme potion geben
- weedkiller potion → zu Dylan bringen
- Devil → von ihm erhält man den wizard slayer trident.
- Wizard slayer trident → damit Zaks stechen
- Zaks → mit Trident stechen - man erhält den Ring von Zaks
- Zaks ring → im Screen links vom Devil auf dem mittleren Fels ablegen
- Moskitos v Vorsicht - kosten Energie

Soviel zu den Tipps und Hinweisen. Ich hoffe, ich habe nichts vergessen Die Grafik ist auch hier wieder mehr als gut gelungen und es macht natürlich Spaß die Umgebung zu erkunden. Natürlich gibt es auch hier wieder den einen oder anderen kleinen Grafikclash, aber das soll dem Spielgeschehen keinen Abbruch tun. Sowas ist man vom Spectrum ja gewohnt. Ebenso - wie auch bei seinen Vorgängern - ist *Magicland Dizzy* kein Spiel, das man eben mal so durchspielt. Man kann sich durchaus eine längere Zeit damit beschäftigen obwohl es einen schon an den Spectrum fesselt - zumindest bei mir war das so. Damit uns das Leben nicht zu leicht gemacht wird, gibt es auch wieder einige Dinge und Kreaturen, die uns an Leder wollen. Besonders knifflig ist es dabei, dass manchmal für den Absprung die Spielfigur pixelgenau zu positionieren ist und man gelegentlich einfach ins Unbekannte springen muss. Aber das macht natürlich auch einen gewissen Reiz aus. Wer mal unser »Betätigungsfeld« in Gänze betrachten möchte, dem habe ich am Ende eine Adresse eingefügt wo man einen Gesamtplan betrachten kann, der erst so richtig deutlich macht, wie groß das Spielfeld dann doch ist. Jetzt wissen wir ja schon mal, auf was wir aufpassen müssen. Nun noch für alle diejenigen, die nicht lange nach den Diamanten suchen wollen, eine gesammelte Aufstellung der Fundorte:

West Tower	Hinter einem Geländer im Sword in the Stone
Zaks Tower	Throne Room
Curiouser and Curriouser	Castle Back Door
Chess Board	Hinter einem Geländer im Bushy Grove
Main Hall	Mysterious Monolith
Entrance Hall	Weirdhenge
Crypt	Haunted Swamp
Oubliette	Unter dem Kessel von Glenda
Troll Bridge	Hot Water Geyser
Top Watch Tower	Hell Gate
The Lower Tallest Tower	Hades
Bards Tree House	Cracks of Gehenna
Tower with a View	Hinter einem Geländer im Cracks of Gehenna
In the Clouds	Down a Well
Up a Tree	Secret Passage

Das also vorab für die ganz Eiligen unter euch.

Kommen wir jetzt zu der schrittweisen Lösung. Wie immer bei *Dizzy-Spielen*, lässt sich auch dieses Programm auf verschiedene Wege lösen. Jeder kann sich seinen Lösungsweg gerne selber zusammenstellen. Es gibt keine einzig Seelig machende Lösung. Aber natürlich sind einige Abhängigkeiten zu beachten, wenn man weiterkommen will. Nachfolgend habe ich euch meinen durchgespielten Weg dargestellt (und hoffentlich dabei nichts vergessen oder falsch notiert), der jedoch wie immer, keinen Anspruch darauf erhebt, das einzig Wahre zu sein. Fangen wir an ...

Wir materialisieren uns in *Weirdhenge* und machen von hier aus wie folgt weiter:

Da es bei den Lösungen von *Dizzy 2* und auch *Dizzy 3* sinnvoll war im ersten Screen nach links zu gehen, machen wir das auch hier und finden am Fuße des *mysteriösen Monolithen* einen *Schlüssel*, den wir sofort aufnehmen. Dann gehen wir noch zwei Screens nach links und kommen an einen *alten Brunnen* über den wir drüber springen (Vorsicht! Nur nicht hineinfallen sonst ist das Spiel hier schon aus) und weiter nach links gehen. Wir stehen am *Burggraben,* in dem ein *Hai* aufgeregt hin- und herschwimmt. Mit dem richtigen Timing springen wir nun auf dessen *Rückenflosse* und versuchen uns laufend/springend nach links an den Eingang zu retten. Wir benutzen den mitgebrachten *Schlüssel* und die Türe öffnet sich. Die eben aufgenommene *Milchflasche* legen wir sogleich wieder ab, denn sie hat keinerlei Nutzen (siehe Aufstellung oben). Wir gehen nach links und finden eine *Kurbel* (handle) die wir aufnehmen, dann im *Treppenhaus* rechts im Screen nach oben und dort wieder 2 Screens nach links. Hier geht es nach oben und wir folgen dem Weg. Hier liegt eine *Power pill* die wir mitnehmen und dann sofort wieder nach unten gehen. Jetzt drei Screens nach rechts. Hier gibt es ein Fenster, vor dem eine *Fledermaus* herumfliegt. Wir springen auf den *Fenstersims* und dann auf die Wolke wobei wir aufpassen, die Fledermaus nicht zu berühren, denn das kostet richtig Energie. Dann springen wir nach rechts und von dieser *Wolke* lassen wir uns nach unten auf das *Dach* des *alten Brunnen* fallen. Von hier springen wir nach rechts auf die *Wolke*

und dann sofort wieder nach links auf die andere Wolke wobei wir den *»cloth duster«* aufnehmen. Dann lassen wir uns wieder nach unten fallen und postieren uns links vom *Brunnen*. Wir benutzen die mitgebrachte *Kurbel* und nehmen dafür den *Eimer* auf. Wir legen den *»cloth duster«* ab und springen über den *Brunnen*. Von hier geht es vier Screens nach rechts bis ans *Wasser*, wo wir die *Geister* hin und herfliegen sehen. Da wir die *Power pill* dabei haben, können wir die *Geister* beseitigen. Wir springen nach rechts auf die *schwimmenden Plattformen* und passen dabei aber auf, dass wir nur kurze Zeit auf ihnen stehen können ehe sie mit uns untergehen. Aber bitte an dieser Stelle keine Panik, mit ein bisschen Geschick ist das kein Problem. Auf der *zweiten Plattform* sehen wir *Dora*, die wir aufnehmen ehe wir weiter nach rechts springen und auf eine *kleine Insel* gelangen auf der die *Hexe Glenda* zu sehen ist. Da wir die *Geister* beseitigt haben und diese nicht mehr erscheinen, können wir die *Power pill* wegwerfen. Wir sprechen mit *Glenda* und erfahren, dass sie schon seit längerer Zeit ihre *Katze* sucht. Von hier aus geht es weiter nach rechts (wieder über schwimmende Plattformen) und wir versuchen uns die *Moskitos* vom Leib zu halten. Weiter nach rechts bis zum *Geysir*, wo wir in der kleinen *Pfütze* unseren mitgeführten *Eimer* mit *heißem Wasser* füllen. Auf dem Rückweg nehmen wir noch die *»drinkme potion«* mit und kehren damit dann bis zur großen *Halle des Schlosses* zurück. Hier legen wir alles ab, was wir bei uns haben nur mit Dora gehen wir einen Screen nach links und geben sie *Prince Charming*. Dann zurück nach rechts und über das *Treppenhaus* nach oben, wieder nach rechts und von hier geht es wieder über die *Wolken* auf das *Dach des Brunnens*. Von hier springen wir zweimal nach rechts und holen uns den dort vorhandenen stick. Dann wieder nach oben und von der Mitte der obersten rechten Plattform aus springen wir weiter nach rechts und finden auf dem Monolithen *Glenda´s schwarze Katze*. Diese wird eingesammelt und am äußersten rechten Rand springen wir weiter nach rechts und landen oben auf *Weirdhenge*. Wir nehmen den *Dolch* und springen von hier aus zweimal nach links um ungeschoren an den *Fackeln* vorbei zu kommen. Dann gehen wir wieder zu *Glenda* und geben ihr die *Katze*. Aus Dankbarkeit, daß wir ihren *Stubentiger* wieder gefunden haben, bietet sie uns an, uns zu helfen. Wir erfahren, was sie benötigt um unseren buschigen Freund zu befreien. Mit diesen Informationen versehen machen wir uns zurück auf den Weg in die *große Halle*. Wenn wir an *Dylan* vorbeikommen, schneiden wir *ein Blatt* von ihm ab, aber wir lassen es jetzt noch liegen. In der *Halle* angekommen nehmen wir den dort deponierten *Eimer* auf und gehen zweimal nach links. Wir sind an der anderen Seite des Burggrabens. Dort gibt. es einen Hebel mit dem wir eine Art *Zugbrücke* einschalten können. Jetzt sind wir in der Lage, den *Burggraben* zu überqueren. Von hier aus geht es wieder dreimal nach links, bis wir an der *Trollbridge* auf eine *Ziege* treffen. Wir schneiden die *Ziege* mit dem *Dolch* los, aber sie bleibt wie angewurzelt stehen. Deshalb braten wir ihr mit dem *Stock* eins über und können sie so überreden, das Weite zu suchen. Auf ihrer Flucht stößt sie auch gleich noch den *Troll* von der *Brücke*, was ja nie schaden kann. Weiter geht es zweimal nach links und hier am Eingang des *Ice Castles* finden wir *Denzil* in einem *Eisblock* eingefroren vor. Wir gießen das heiße Wasser auf ihn und *Denzil* verschwindet. Ehe er das tut überlässt er uns noch seinen *Walkman*. Damit geht es zurück in den große Halle, wo wir den *Walkman* ablegen. Dafür greifen wir uns die *»drinkme potion«* und steigen die Stufen abermals nach oben. Wieder springen wir auf den *Fenstersims* und von hier nach links oben auf das *Dach des Bettes*. Ganz links am *Bett* springen wir nun nach links und kommen zu einem *Geländer*. Hier geht es nach oben und wir tun dies zwei Screens weit. Hier liegt der *»lightning rod«*, den wir aufnehmen, dann nichts wie runter und nach links. Hier greifen wir uns das *»gold cross«* und weiter nach links. Hier ist nun ein offenes Fenster zu sehen aus dem wir springen. Wir gehen nach links und stellen den *»lightning rod«* bei *Dozy* ab. Von hier aus geht es nun 3 Screens nach links und dann zweimal nach unten. Durch das *Kreuz*, weicht uns der *Vampir* aus und wir können unseren Weg fortsetzen. Es ist jedoch unmöglich der dort herumfliegenden *Fledermaus* auszuweichen. Doch wenn wir uns nicht zu ungeschickt anstellen, wird sie uns nur kurz berühren und nicht allzu viel Schaden anrichten. Hinter einer Treppe ist eine Fackel zu sehen, die wir aufnehmen. Dadurch öffnet sich eine *Geheimgang* ganz rechts unten und wir können hier nach rechts zu Daisy hindurchschlüpfen. Wir benutzen die mitgebrachte „*drinkme potion*" und Daisy ist befreit. Dann geht. es zurück nach links, zweimal nach oben und viermal nach rechts. Wir nehmen den *Apfel* auf und legen dafür das *Kreuz* ab. Dann gehen wir nach rechts weiter bis zu *Dylan* und nehmen das vorhin *abgeschnittene Blatt* auf. Mit diesen drei Sachen (*Fackel*, *Apfel*, *Blatt*) machen wir uns wieder auf den Weg zu *Glenda* und geben ihr alle drei Dinge. Vorher ist aber unbedingt darauf zu achten, dass wir den *Diamanten* unter dem *Kessel* entfernt haben, da wir sonst nicht mehr rankommen, wenn das Feuer brennt. *Glenda* braut uns nun einen *Unkrautvernichtungszaubertrank* (was für ein Wort) und damit kehren wir zu *Dylan* zurück. Wir wenden das Mittel an und *Dylan* ist ebenfalls befreit. Wir gehen zur *großen Halle* und nehmen unterwegs den *»cloth duster«* mit. Dort angekommen legen wir ihn wieder ab und greifen uns den *Walkman*. Jetzt wieder die *Treppe* hoch (Kondition ist gefragt in diesem Spiel) und zweimal nach links. Es folgt ein erneuter *Sprung aus dem Fenster* und wir gehen nach unten. Vom äußersten linken *Rand der Plattform* springen wir nach links, von der Mitte *der Plattform* aus springen wir nach rechts und hier wieder nach links. So erreichen wir die *Hütte des Barden*, der dort unbeeindruckt auf seiner *Flöte* einen abbläst. Wir geben ihm den *Walkman* und der Typ rastet förmlich aus. Er wirft seine *Panflöte* weg und verschwindet. Wir nehmen die *Flöte* auf und gehen bis zum *Brunnen*. Wir stellen uns auf die *rechte* Seite *(wichtig!!!)* und lassen uns nach links in den *Brunnenschacht* fallen. Wir bleiben vor einem Eingang stehen, der durch eine *riesige Ratte v*ersperrt wird. Deshalb springen wir nach rechts auf den kleinen *Mauervorsprung* und spielen dort mit der *Flöte* ein kleines Liedchen. Nach einiger Zeit erwacht die *Ratte* zu Leben und geht auf uns los. Da wir aber ganz rechts stehen, erreicht sie uns nicht und fällt in den *Brunnenschacht*. Die wären wir los. Wir springen wieder nach links und

weiter geht´s in dieser Richtung bis wir zu etwas *»klebrigem«* kommen das wir mitnehmen. Weiter geht es nach links und wir finden eine antike *Lampe*, die wir ebenfalls aufnehmen. Vorsicht vor den *Säuretropfen* die von der Decke fallen, da kann man schnell ein Leben aushauchen. Wir springen nach oben und legen die *Pipes* ab. Dafür nehmen wir den *»cloth duster«* mit. Wir gehen zum *Schwert im Stein* und ziehen es heraus. Eine Meldung sagt uns, dass wir nun *King Dizzy* sind. Wir gehen nun nach links zu *Dozy* und legen die *Lampe* in seiner Nähe ab. Dann benutzen wir *den »cloth duster«* und *Zaks* erscheint. Dadurch, dass wir den *lightning rod* direkt vor *Dozy* aufgestellt haben, springt *Zaks* direkt darauf über und erschreckt *Dozy*. Deshalb wacht er aus seinem *Tiefschlaf* auf und kann ebenfalls nach Hause gehen. Nun hätten wir ja schon fast alle unsere Familienmitglieder befreit, fehlt nur noch *Grand Dizzy*. Aber den werden wir auch noch finden. Wir gehen ins *Ice Castle* und springen hier nach links oben. Von hier aus zweimal nach oben und hier springen wir am äußersten linken Punkt der oberen linken Plattform wiederum nach links. Nach diesem weiten Sprung gehen wir nach unten und sehen nun an der linken Wand einen Spiegel. Wir springen hindurch und alles geht anders herum. Das kostet etwas Gewöhnung aber man kommt damit zurecht. Wir sind auf dem Schachbrett und da wir *King Dizzy* sind, können wir die *Königin* schlagen. Hätten wir das *Schwert Excalibur* nicht bei uns wären wir ein *Bauer* und die *Königin* würde uns sofort schlagen. Da haben wir eben noch mal Glück gehabt. Wir gehen weiter nach links und passen dabei auf, nicht mit den *Bauern* zu kollidieren, die uns sofort die Energie entziehen würden. Wir befreien *Grand Dizzy*, der hier etwas unbeholfen herumsteht und nicht weiß, wie er hierhergekommen ist. Aber der Typ schlafwandelt ja gerne. Von hier aus geht es jetzt an das ganz *entgegengesetzte Ende* von *Magicland*, an den *Vulkankrater*. Nachdem wir alle unsere Familienmitglieder gerettet haben steigen nun aus dem Krater *kleine Wölkchen* auf mit deren Hilfe wir den *Krater* überqueren können. Weiter geht´s nach rechts und dann zweimal nach unten. Rechts unten in der Ecke steht der *Teufel* umgeben von einem Kranz mit Flammen. Wir versuchen den *hin- und herlaufenden Flammen* bestmöglich auszuweichen und treten vor den *Teufel*. Wir bekommen einen *Dreizack* mit dem wir nun wiederum zurück uns *Ice Castle* müssen um Zaks damit zu vertreiben. Deshalb müssen wir dort ganz nach oben und hier ist es sinnvoll im Raum mit Zaks möglichst weit rechts zu stehen, so dass uns die Blitze nicht gefährlich werden können. Im richtigen Moment springen wir dann nach links und stechen *Zaks* mit dem *Dreizack*, was ihn verschwinden lässt. Nur sein *Ring* bleibt zurück und den nehmen wir an uns. Damit dann wieder zurück zum *Teufel* und unten links aus dem Screen heraus. Wir springen auf den *mittleren Felsen* und werfen den Ring an der *linken Seite* in die *Lava*. Damit ist *Zaks* endgültig besiegt. Inzwischen sollten wir aber alle *30 Diamanten* eingesammelt haben und so können wir abermals vor den *Teufel* treten. Sein *Feuerkranz* ist nun beseitigt und er ermöglicht uns einen *Teleport* nach Hause. Der Preis dafür, wie könnte es anders sein, beträgt exakt *30 Diamanten*. So ein Zufall aber auch. Nun gut, jedenfalls haben wir damit *Magicland Dizzy* erfolgreich beendet. Hoffentlich hattet ihr so viel Spaß dabei wie ich und ihr seid mit der vorhandenen Energie und den Leben ausgekommen. Ich jedenfalls habe dazu einige Anläufe benötigt.

Ganz zum Schluss noch ein paar Hinweise für diejenigen, die ein wenig »tricksen« wollen. Nachfolgend aufgeführte Pokes habe ich dem Internet entnommen jedoch selbst nicht ausprobiert, weil mir sonst der Reiz am Spiel fehlt. Wenn man den Quellen trauen darf, handelt es sich dabei teilweise um Multiface-Pokes. Jetzt möchte ich diese hier noch schnell vollständigkeitshalber - aber ohne Gewähr - auflisten:

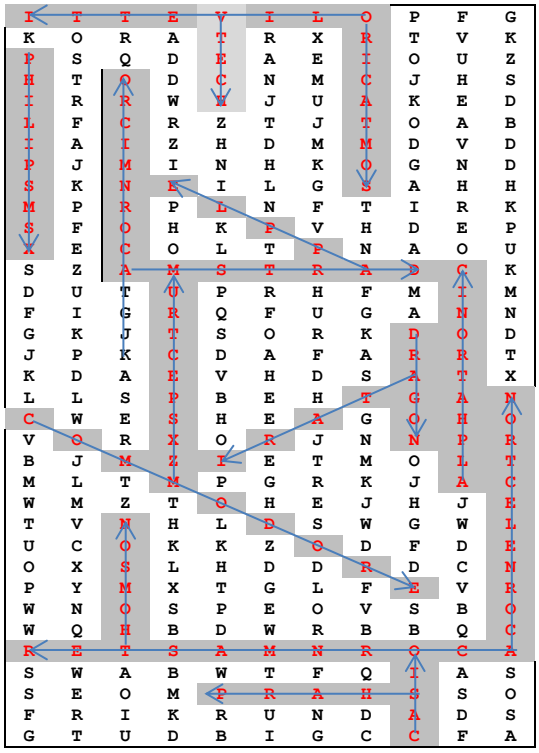
Unendliche Leben:	**29623,182**
Durch Mauern laufen:	**41179,201**
Spiel beschleunigen:	**405436,1**
Immun gegen Feuer:	**35903,70**
Immun gegen Sumpfgeister:	**33020,70**
Immun gegen Ratte:	**33085,70**
Immun gegen Fledermäuse:	**32593,70**

Wer einen Blick auf den Gesamtplan werfen möchte der sei an dieser Stelle auf die Speccy Screenshot Maps **https://maps.speccy.cz/map.php?id=Dizzy4&sort=0&part=99&ath=0** verwiesen.

Und nun wünsche ich euch noch viel Spaß und Glück beim Spielen von *Dizzy 4 - Magicland*. Wert ist es das Programm auf jeden Fall.

© 2022 Harald Lack

LÖSUNG PUZZLERÄTSEL



TASTATURFOLIEN FÜR RETRO-COMPUTER

VON THOMAS EBERLE

Bereits vor annähernd 30 Jahren war die erste größere Investition von SINTECH die Produktion von Tastaturfolien. Das war auch sehr nötig, hatten doch die in den 80er Jahren verbauten Folien von Sinclair nur eine Haltbarkeit von ungefähr 10 Jahren und fielen reihenweise aus. Als, soweit uns bekannt ist, einziger Hersteller weltweit produziert SINTECH die Tastaturfolien für Sinclair QL und das bis heute. SINTECH ist nicht der einzige, ist aber bei den Folien für ZX Spectrum (Gummi) und ZX Spectrum + / 128, unter den Reparaturdienstleistern sehr beliebt da sehr genau, funktional und außerdem langlebig.

Amstrad hatte die Fehler von Sinclair nicht wiederholt und in die späteren Spectrum-Geräte und auch in die Amstrad CPC Geräte sehr gute Folien eingebaut. Trotzdem sind diese Geräte, ebenso wie die ersten AMIGA Computer, fast 40 Jahre alt, und aus verschiedenen Gründen können sie dennoch ausfallen. Diesem wachsendem Bedarf ist SINTECH ebenfalls begegnet und hat Tastaturfolien für diese Computer hergestellt.

Folgende Modelle erhalten Sie heute bei SINTECH:

- ZX81
- ZX Spectrum Gummi
- ZX Spectrum +/128
- Amstrad CPC 6128
- Amstrad CPC Plus 464 / 6128
- Amiga 500 (alle Modelle)
- Amiga 600 (grüne und blaue Folie)
- Amiga 1200

In Planung:

- Atari ST
- Amstrad CPC-Reihe
- Acorn BBC Micro

Die meisten Folien sind selbst entwickelt, dabei hat SINTECH z.B. beim Amiga 500 eine universelle Folie entwickelt, wo vorher verschiedene Folien benötigt wurden.

Die Entwicklung geht weiter. Je älter die Computer werden, desto größer wird die Nachfrage nach nach Tastaturfolien. Erste Tests laufen bereits für Acorns BBC Micro, aber auch Atari ST, Amstrad CPC 464/664 sind bei SINTECH in der Planung.

SINTECH merkt hier an, dass es natürlich Lücken in Bedarf der Tastaturfolien für Retro-Computer gibt. Originalzitat von Herr SINTECH: "Natürlich können wir aber nicht alles wissen. Gibt es Tastaturfolien die Sie in unserem Lieferprogramm vermissen? Wir bekommen gerne Ihr Feedback und kümmern uns darum"

Herr SINTECH ist für jede Rückmeldung sehr dankbar, da es ihm ein Anliegen ist, die Computer, auf denen die moderne Computertechnik basiert, am Leben und für die Nachwelt zu erhalten.

Schreiben Sie ihm. Er freut sich darüber:

info@sintech-shop.de

Retro-Werbung 1983

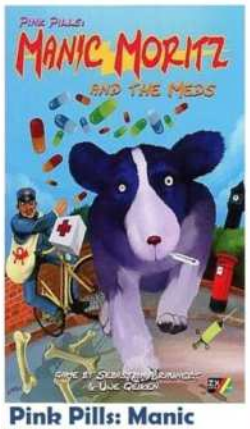
sintech

www.sintech-shop.de

Ersatzteile, Zubehör und Reparaturen
für Mobiltelefone, Spielekonsolen und mehr

SINTECH GmbH
Kurmainzstr. 14 - 74386 Cleebronn

Supermoritz



Pink Pills: Manic Moritz and the meds

Jonathan Trick or Treat by Team Moritz

Mecki Winter Hibernation

Moritz the Striker

Cheesy Chase: Moritz and the mildewed moon

Moritz on the Autobahn

Neue Spiele für den ZX Spectrum! Alle Infos und Preise auf
www.sintech-shop.de/retro-atari-commodore-sinclair-etc/sinclair/spectrum/neue-software

sintech DEUTSCHLAND
sintech CZECH REPUBLIC
sintech UNITED KINGDOM